OIPE
OCT 12 1999
PATENT & TRADEMARK OFFICE
JC90

1/5/1 (Item 1 from file: 347)
DIALOG(R)File 347:JAPIO
(c) 1999 JPO & JAPIO. All rts. reserv.

04587521 **Image available**
DOCUMENT PROCESSOR

PUB. NO.: 06-259421 **JP 6259421** A]
PUBLISHED: September 16, 1994 (19940916)
INVENTOR(s): KOJO SHINTARO
MIYAZAKI ATSUSHI
MATSUMOTO TEN
APPLICANT(s): FUJI XEROX CO LTD [359761] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 05-164761 [JP 93164761]
FILED: July 02, 1993 (19930702)
INTL CLASS: [5] G06F-015/20; G06F-015/20; G06F-015/40
JAPIO CLASS: 45.4 (INFORMATION PROCESSING -- Computer Applications)
JAPIO KEYWORD: R131 (INFORMATION PROCESSING -- Microcomputers & Microprocessers)
JOURNAL: Section: P, Section No. 1843, Vol. 18, No. 659, Pg. 126, December 13, 1994 (19941213)

ABSTRACT

PURPOSE: To provide the document processor which can execute a retrieval processing of a document component confirming to a designated hierarchical structure pattern to a structuralized document.

CONSTITUTION: In a memory 10, pattern described information 20 for showing the information in which a connecting relation of each document is described is stored, and in a document file 60, a structuralized document is stored. An interpreting part 30 interprets the pattern described information 20, generates a document structure pattern for expressing a hierarchical structure, and stores this document structure pattern 40 in the memory 10. A reorganizing part 50 scans the structuralized document of the document file 60, reorganizes it to a structure of a format which can execute a collation processing, and stores a reorganized structuralized document 70 being its result in the memory 10. A collating part 80 collates the document structure pattern 40 and the reorganized structuralized document 70. An output processing part 90 outputs that which is coincident by a collation of the collating part 8.

NE-840

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平6－259421

(43)公開日　平成6年（1994）9月16日

(51)Int.Cl.[5]	識別記号	庁内整理番号	Ｆ Ｉ	技術表示箇所
Ｇ０６Ｆ　15/20	５５０　Ｅ	7315－5Ｌ		
	５５４　Ｍ	7315－5Ｌ		
15/40	５００　Ｌ	9194－5Ｌ		

審査請求　未請求　請求項の数７　ＯＬ（全 33 頁）

(21)出願番号　特願平5－164761

(22)出願日　平成5年（1993）7月2日

(31)優先権主張番号　特願平４－１７６７９２

(32)優先日　平4（1992）7月3日

(33)優先権主張国　日本（ＪＰ）

(31)優先権主張番号　特願平５－２８５５

(32)優先日　平5（1993）1月11日

(33)優先権主張国　日本（ＪＰ）

(71)出願人　000005496
富士ゼロックス株式会社
東京都港区赤坂三丁目3番5号

(72)発明者　古城　慎太郎
神奈川県川崎市高津区坂戸3丁目2番1号　ＫＳＰ　Ｒ&Ｄ　ビジネスパークビル　富士ゼロックス株式会社内

(72)発明者　宮崎　淳
神奈川県川崎市高津区坂戸3丁目2番1号　ＫＳＰ　Ｒ&Ｄ　ビジネスパークビル　富士ゼロックス株式会社内

(74)代理人　弁理士　木村　高久

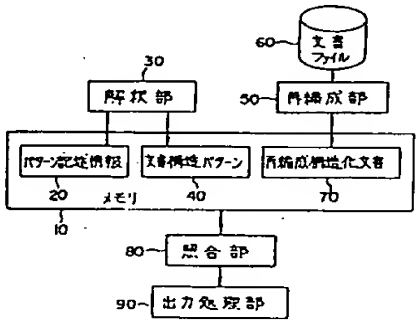
最終頁に続く

(54)【発明の名称】文書処理装置

(57)【要約】

【目的】構造化文書に対する指定された階層構造パターンに従った文書構成要素の検索処理を行うことのできる文書処理装置を提供する。

【構成】メモリ１０には、文書構造要素同志の接続関係が記述された情報を示すパターン記述情報２０が格納され、また文書ファイル６０には構造化文書が格納されている。解釈部３０は、パターン記述情報２０を解釈して、階層構造を表現している文書構造パターンを生成し、この文書構造パターン４０をメモリ１０に格納する。再編成部５０は、文書ファイル６０の構造化文書を走査して、照合処理可能な形式の構造に再編成し、この結果である再編成構造化文書７０をメモリ１０に格納する。照合部８０は文書構造パターン４０と再編成構造化文書７０とを照合する。出力処理部９０は、照合部８０の照合により一致したものを出力する。



1

【特許請求の範囲】

【請求項1】複数の文書構成要素を有する構造化文書に対する処理を行う文書処理装置において、

基準となる文書構成要素同志の接続関係を解釈する解釈手段と、

前記構造化文書を照合処理可能な形式の構造に再編成する再編成手段と、

前記解釈手段による解釈結果と、前記再編成手段による再編成結果とを照合する照合手段と、

この照合手段の照合により一致した文書構成要素を前記再編成結果から抽出し出力する出力処理手段とを具備したことを特徴とする文書処理装置。

【請求項2】所定の文書構成要素に対する所定の処理を指定する指定手段を更に具備し、

前記出力処理手段は、前記照合手段の照合により一致した文書構成要素に対して、前記指定手段により指定された所定の処理を施した後、出力することを特徴とする請求項1記載の文書処理装置。

【請求項3】所定の文書構成要素に対する削除処理を指定する指定手段と、

前記照合手段の照合により一致した文書構成要素と、当該文書構成要素の親である文書構成要素とを対応付けして蓄積する蓄積手段とを更に具備し、

前記出力処理手段は、前記指定手段により削除処理が指定されると、前記蓄積手段に蓄積されている前記親である文書構成要素から、前記蓄積手段に蓄積されている前記照合手段の照合により一致した文書構成要素に関する情報を取り除くと共に、当該親である文書構成要素から削除されない、当該親の文書構成要素の子供である文書構成要素を出力することを特徴とする請求項1記載の文書処理装置。

【請求項4】複数の文書構成要素を有する構造化文書を複数保存したファイルを格納する格納手段と、

基準となる文書構成要素同志の接続関係を解釈する解釈手段と、

前記格納手段に格納されているファイル内の複数の構造化文書それぞれを照合処理可能な形式の構造に再編成する再編成手段と、

前記解釈手段による解釈結果と、前記再編成手段による再編成結果とを照合する照合手段と、

この照合手段の照合により一致した文書構成要素を前記再編成結果から抽出し出力する出力処理手段とを具備したことを特徴とする文書処理装置。

【請求項5】前記複数の構造化文書を保存した複数のファイルを対象として、前記文書構成要素同志の接続関係に適合する文書構成要素を前記再編成結果から抽出することを特徴とする請求項4記載の文書処理装置。

【請求項6】ソース側及びターゲット側それぞれの前記複数の構造化文書を保存した複数のファイルを対象として、前記文書構成要素同志の接続関係に適合する文書構成要素を前記再編成結果から抽出すると共に、前記ターゲット側の抽出結果である文書構成要素に対する前記ソース側の抽出結果である文書構成要素の挿入を実行することを特徴とする請求項4記載の文書処理装置。

【請求項7】前記出力処理手段は、指定された属性に関する情報に基づいて、前記照合手段の照合により一致した文書構成要素の属性の参照又は変更の操作を実行することを特徴とする請求項2乃至請求項6記載の文書処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は、文書中から所望のパターンを検索する文書処理装置に関する。

【0002】

【従来の技術】従来においては、ワードプロセッサや、文書作成・編集機能を有するワークステーションやコンピュータ等の装置には、一般的に、作成された文書中から所望の文字列を検索することのできる検索機能が設けられている。この検索機能を利用することにより、検索対象となる文字列を、他の文字列に置換したり削除したりして、文書を編集することができる。

【0003】またこのような文書編集を自動化するようにしたものも実現されており、例えば、

（1）カット・アンド・ペーストしながらの操作をマクロ記述して何度でも実行することができるようにしたもの。

（2）例えば特開平3－147062号公報に開示されている様に、文書中から取り出された複数の文字列を一時記憶領域に保存し、その後順次、ペースト（挿入）するようにしたもの。

（3）指定したパータン文字列が文字列中に現れた際に、置き換え等を実施する様にしたもの（例えばUNIXのｓｅｄのようなストリームエデッタ）。がある。

【0004】また文書ファイルを編集する方法としては、インタラクティブにユーザがカット・アンド・ペーストする方法や、バッチ処理で行う方法がある。このうち、効率良く複数回の抽出挿入を行う方法として、例えば、

（4）ユーザが指定した抽出文字列を複数個同時に記憶しておき、これら複数の文字列を順次、所定の位置に挿入するようにしたもの（例えば特開平3－147062号公報）。

（5）ユーザがインタラクティブに文書にマークを付与することにより、一度に複数の指定領域の入れ替えを行うようにしたもの（例えば特開平3－260761号公報）。

（6）ファイル内のある特定の文字列パターンの出現に対して、文字列の置き換え等の処理を行うようにしたもの（例えばUNIXのａｗｋ、ｓｅｄなど）。

（7）構造と内容とを分離して管理する構造化文書シス

テムにおいて、差し込み指定のある複数の文書から、差し込み文字列を予め抽出しておき、この差し込み文字列を差し込み対象文書へ反映させるようにしたもの（例えば特開平4－57151号公報）。がある。

【0005】更に文書の特定部分の属性値（例えば文字の大きさなど）を変更するようにしたものも実現されており、この方法として、例えば、

（8）インタラクティブなエディタにより逐一変更するようにしたもの。

（9）特定部分にスタイルを予め設定しておき、スタイルの属性値を変更することにより、一括変換するようにしたもの。

がある。

【0006】

【発明が解決しようとする課題】しかしながら、上記従来の装置では、文書中から文字パターン（検索対象の文字列）を検索することは可能であるが、これは文書中の文字列と指定文字列との照合により一致したものを検索するようにしているので、内部的に階層構造の情報を持った文書いわゆる構造化文書から、指定された階層構造パターンに従った文書要素を検索することができなかった。

【0007】すなわち、構造化文書においては、文書要素が階層構造上のどこに位置するかによって、その文書要素の情報は意味を変えるので、従来の如く、単に文字列の照合のみにより検索し、階層構造を考慮しない検索では、適切な部分にアクセスすることができず、正確な検索処理を行うことができなかった。

【0008】また上記（1）のものでは、不正確な処理を施してしまうことが多く、このため、複数の文書をバッチで処理することができない。

【0009】また上記（2）のものでは、正確な処理を施すことはできるが、バッチで処理することができない。

【0010】また上記（3）のものでは、バッチ処理に適しているが、構造を持った文書の文字列は構造上の位置によって意味を変えるので、不適切な部分を書き換えてしまう恐れがあり、構造化文書には適していない。

【0011】また上記（4）のものでは、文書中の全指定箇所をユーザがインタラクティブに一度設定しなければならないので、操作が繁雑である。

【0012】また上記（5）のものでは、上記（4）の場合と同様に、文書中の全指定箇所をユーザがインタラクティブに一度設定しなければならないので、操作が繁雑である。

【0013】また上記（6）のものでは、ユーザが文書中の所定箇所を指定する必要はないが、一般的なストリングマッチのため、ファイル中の構造の意味を解釈せずに、単に文字列として照合し、一致した文字列に対して変更を加えるようにしているので、文書の構造の意味を維持しつつ処理することはできない。

【0014】また上記（7）のものでは、内容の変更については効率良く実行することができるが、構造と内容を同時に変更することはできない。

【0015】また上記（8）の方法では、インタラクティブに処理するので、属性値の変更処理を自動化することができず、このため効率が悪く、操作ミスによる間違いも発生し易い。

【0016】更に上記（9）のものでは、予めスタイルを設定しておく必要であるので、そのための手間がかかる。また同一のスタイルには全て同一の属性が付与されるので、柔軟性に欠ける。

【0017】そこで本発明の目的は、構造化文書に対する指定された階層構造パターンに従った文書構成要素の検索処理を行うことができる文書処理装置を提供することである。

【0018】本発明の他の目的は、構造化文書に対する指定された階層構造パターンに従った文書構成要素を検索し、この文書構成要素に対する削除、置換、複写などの操作を実施することができる文書処理装置を提供することである。

【0019】本発明の他の目的は、複数の構造化文書に対する指定された階層構造パターンに従った文書構成要素の検索処理を行うことができる文書処理装置を提供することである。

【0020】本発明の他の目的は、ターゲット側の複数のファイルそれぞれに保存されている複数の構造化文書から抽出された文書構成要素に対する、ソース側の複数のファイルそれぞれに保存されている複数の構造化文書から抽出された複数の文書構成要素の挿入操作を一度に自動的に行うことができる文書処理装置を提供することである。

【0021】本発明の他の目的は、複数の構造化文書に対する指定された階層構造パターンに従った文書構成要素を検索し、この文書構成要素の属性値の参照又は変更を行うことができる文書処理装置を提供することである。

【0022】

【課題を解決するための手段】第1の発明は、複数の文書構成要素を有する構造化文書に対する処理を行う文書処理装置において、基準となる文書構成要素同志の接続関係を解釈する解釈手段と、前記構造化文書を照合処理可能な形式の構造に再編成する再編成手段と、前記解釈手段による解釈結果と、前記再編成手段による再編成結果とを照合する照合手段と、この照合手段の照合により一致した文書構成要素を前記再編成結果から抽出し出力する出力処理手段とを具備している。

【0023】第2の発明は、第1の発明において、所定の文書構成要素に対する所定の処理を指定する指定手段を更に具備し、前記出力処理手段は、前記照合手段の照

合により一致した文書構成要素に対して、前記指定手段により指定された所定の処理を施した後、出力することを特徴としている。

【００２４】第３の発明は、第１の発明において、所定の文書構成要素に対する削除処理を指定する指定手段と、前記照合手段の照合により一致した文書構成要素と、当該文書構成要素の親である文書構成要素とを対応付けして蓄積する蓄積手段とを更に具備し、前記出力処理手段は、前記指定手段により削除処理が指定されると、前記蓄積手段に蓄積されている前記親である文書構成要素から、前記蓄積手段に蓄積されている前記照合手段の照合により一致した文書構成要素に関する情報を取り除くと共に、当該親である文書構成要素から削除されない、当該親の文書構成要素の子供である文書構成要素を出力することを特徴としている。

【００２５】第４の発明は、複数の文書構成要素を有する構造化文書を複数保存したファイルを格納する格納手段と、基準となる文書構成要素同志の接続関係を解釈する解釈手段と、前記格納手段に格納されているファイル内の複数の構造化文書それぞれを照合処理可能な形式の構造に再編成する再編成手段と、前記解釈手段による解釈結果と、前記再編成手段による再編成結果とを照合する照合手段と、この照合手段の照合により一致した文書構成要素を前記再編成結果から抽出し出力する出力処理手段とを具備している。

【００２６】第５の発明は、第４の発明において、前記複数の構造化文書を保存した複数のファイルを対象として、前記文書構成要素同志の接続関係に適合する文書構成要素を前記再編成結果から抽出することを特徴としている。

【００２７】第６の発明は、第４の発明において、ソース側及びターゲット側それぞれの前記複数の構造化文書を保存した複数のファイルを対象として、前記文書構成要素同志の接続関係に適合する文書構成要素を前記再編成結果から抽出すると共に、前記ターゲット側の抽出結果である文書構成要素に対する前記ソース側の抽出結果である文書構成要素の挿入を実行することを特徴としている。

【００２８】第７の発明は、第２の発明乃至第６の発明において、前記出力処理手段は、指定された属性に関する情報に基づいて、前記照合手段の照合により一致した文書構成要素の属性の参照又は変更の操作を実行することを特徴としている。

【００２９】

【作用】第１の発明では、解釈手段によって解釈された文書構造パターンと、再編成手段によって再編成された構造化文書とを照合手段によって照合し、更に出力処理手段が、その照合により一致した文書構成要素を再編成された構造化文書から抽出し出力するようにしたので、構造化文書から、指定された階層構造に従った文書構成要素を検索し出力することができる。

【００３０】第２の発明では、第１の発明において、出力処理手段は、照合手段の照合により一致した文書構成要素に対して、指定手段により指定された所定の処理例えば削除、置換、複写などの処理を施した後、出力するようにしているので、構造化文書から、指定された階層構造に従った文書構成要素に対して、削除、置換、複写などの処理を施すことができる。

【００３１】第３の発明では、第１の発明において、出力処理手段は、指定手段により削除処理が指定されると、蓄積手段に蓄積されている、照合手段の照合により一致した文書構成要素の親である文書構成要素から、蓄積手段に蓄積されている照合手段の照合により一致した文書構成要素に関する情報（例えば文書構成要素を示すノード、そのノードの位置情報）を取り除くと共に、当該親である文書構成要素から削除されない、当該親の文書構成要素の子供である文書構成要素を出力するようにしているので、削除すべき文書構成要素の親の文書構成要素の内容を自動的に変更することができる。

【００３２】第４の発明では、解釈手段が、基準となる文書構成要素同志の接続関係を解釈し、また再編成手段が、格納手段に格納されているファイル内の複数の構造化文書それぞれを照合処理可能な形式の構造に再編成し、また照合手段が、解釈手段による解釈結果と、再編成手段による再編成結果とを照合し、更に出力処理手段が、照合手段の照合により一致した文書構成要素を前記再編成結果から抽出するようにしているので、複数の構造化文書から、指定された階層構造に従った文書構成要素を検索し出力することができる。

【００３３】第５の発明では、第４の発明において、複数の構造化文書を保存した複数のファイルを対象として、文書構成要素同志の接続関係に適合する文書構成要素を再編成結果から抽出するようにしているので、複数のファイルそれぞれに保存されている複数の構造化文書から、指定された階層構造に従った文書構成要素を検索し出力することができる。

【００３４】第６の発明は、第４の発明において、ソース側及びターゲット側それぞれの複数の構造化文書を保存した複数のファイルを対象として、文書構成要素同志の接続関係に適合する文書構成要素を再編成結果から抽出すると共に、ターゲット側の抽出結果である単数又は複数の文書構成要素に対するソース側の抽出結果である単数又は複数の文書構成要素の挿入を実行するようにしているので、ターゲット側における複数のファイルそれぞれに保存されている複数の構造化文書から抽出された単数又は文書構成要素に対して、ソース側における複数のファイルそれぞに保存されている複数の構造化文書から抽出された単数又は文書構成要素を一度に挿入することができる。

【００３５】第７の発明では、第２の発明乃至第６の発

明において、出力処理手段は、指定された属性に関する情報に基づいて、照合手段の照合により一致した文書構成要素の属性の参照又は変更の操作を実行するようにしているので、構造化文書から、指定された階層構造に従った文書構成要素を検索し、この文書構成要素の属性に対する参照又は変更の操作を実施することができる。

【００３６】

【実施例】以下、第１の実施例乃至第５の実施例を添付図面を参照して説明する。

【００３７】最初に第１の実施例を図１乃至図９を参照して説明する。

【００３８】図１は、本発明に係る文書処理装置の第１の実施例を示す機能ブロック図である。

【００３９】同図において、メモリ１０には、基準となる文書構成要素同志の接続関係（階層関係や順序関係）のパターン記述情報２０（これについては後述する）が記憶されており、解釈部３０は、メモリ１０からパターン記述情報２０を読み出して解釈し、この解釈結果である文書構造パターン４０（これについては後述する）をメモリ１０に記憶する。再編成部５０は、文書ファイル６０に保存されている構造化文書内を走査して、その構造化文書を照合処理可能な形式の構造に再編成し、この再編成結果である再編成構造化文書７０（これについては後述する）をメモリ１０に格納する。そして照合部８０は、メモリ１０に記憶されている文書構造パターン４０と再編成構造化文書７０と照合し、この照合結果を出力処理部９０に出力する。出力処理部９０では、照合部８０の照合により一致した文書構成要素を再編成構造化文書７０から抽出し出力する。

【００４０】図２は、図１に示した実施例の装置を実現するためのハードウェア構成を示したものであり、例えば、ワークステーションやコンピュータ等のブロック図を示している。図２において、装置は、構造化文書に対する検索処理を実行する中央処理装置（以下、ＣＰＵという）２１０と、主メモリ２２０と、ディスク２３０と、各種のデータ内容や文書内容を表示するディスプレイ２４０と、キーボードやマウスから構成され各種データや指令を入力する入力装置２５０とがバス２６０を介してそれぞれ接続されている。なおＣＰＵ２１０はバス２６０を介してこれに接続された各部を制御する。

【００４１】ここで、図１に示した機能ブロック図の構成要素と図２に示したブロック図の構成要素との対応関係について説明する。図１に示したメモリ１０は主メモリ２０に対応しており、図１に示した解釈部３０、再編成部５０、照合部８０、および出力処理部９０は共にＣＰＵ２１０に対応しており、文書ファイル６０はディスク２３０に対応している。

【００４２】次に上述したパターン記述情報２０について説明する。

【００４３】パターン記述情報２０は、パターンが、“節”、“表題”などの単純文字列パターン、“／”、“＃”などの接続表現記号、“（”、“）”などの括弧、等の構文要素が特定の文法に従って出現するように表現されるものであり、例えば、

（節／表題／ｉｎｔｒｏｄｕｃｔｉｏｎ）＃（節／表題）　…（１）

ここで、“／”は包含関係、“＃”は順序関係を表している。

のように記述される。この記述内容はメモリ１０に記憶される。

【００４４】そして解釈部３０は、メモリ１０から上記（１）に示す様なパターン記述情報を読み出して解釈し、この結果として図３に示す様に階層構造（木構造）を形成している文書構造パターン（これが上述した文書構造パターン４０に相当する）を生成する。このとき、当然、上述したような構文要素や文法などが考慮されて、文書構造パターンが生成されることとなる。なおこの実施例では、図３に示すような文書構造パターンにおける矩形図形を単純パターンということにする。

【００４５】ここで、解釈部３０によるパターン記述の解釈処理について、図４に示すフローチャートを参照して説明する。なおここでは、括弧（“（”、“（”）の構文要素の処理を省略している。

【００４６】解釈部３０は、カレントレコードを生成し（ステップ４０１）、その後、入力文字列（例えば上記（１）のパターン記述情報）についての解釈は終りか否かを判断する（ステップ４０２）。入力文字列についての解釈処理がまだ残っている場合は、次の文字が順序関係を表す記号“＃”か否かを判断する（ステップ４０３）。

【００４７】ステップ４０３において記号“＃”であれば、新たなレコードを生成し、このレコードをカレントレコードの弟にし（ステップ４０４）、その後、新たに生成したレコードをカレントレコードと定義する（ステップ４０５）。その後、上記ステップ４０２に戻りこのステップ以降を実行する。

【００４８】ステップ４０３において文字が記号“＃”でない場合は、当該文字が包含関係を表す記号“／”か否かを判断し（ステップ４０６）、記号“／”の場合は、新たなレコードを生成し、このレコードをカレントレコードの子にする（ステップ４０７）。その後、ステップ４０５に進む。

【００４９】ステップ４０６において文字が記号“／”でない場合は、文字であることを意味するので、当該文字をカレントレコード内に挿入し（ステップ４０８）、その後、ステップ４０２に戻りこのステップ以降を実行する。

【００５０】なお、ステップ４０２において入力文字列についての解釈が終了した場合は処理を終了する。

【００５１】ここで、具体例を挙げて説明する。例えば

“富士夫／太郎＃花子”という文字列は、図5（a）～（j）に示すようにパターンとして解釈されていく。なお、同図において、矩形図形がレコードを表している。また図5（j）に示す内容が最終的な文書構造パターンである。

【0052】上述した例では、文字列のパターン解釈であったが、図形エディタを用いて描画したグラフ（グラフ理論におけるグラフ）を解釈してパターンとする方法もある。このときは、ノードやリンクを適切な意味に解釈するように定義する。例えば、図3に示した例では、矩形で囲まれた文字列（例えば節や表題）が単純文字列パターンを表し、上下の矩形図形を結んでいるリンク（例えば符号301で示す線分（リンク））が序列関係を表し、左右の矩形図形を結んでいるリンク（例えば符号302で示す線分（リンク））が包含関係を表している。このような図形から意味構造を抽出するには、例えばパターン記述専用の図形エディタを用意すれば良い。

【0053】ここで、図形エディタを用いたパターン記述の一例を図6に示す。

【0054】まずユーザは、図6（a）に示す様にパターンエディタの初期画面つまりウィンドウ600を表示画面に表示せさ、次に図6（b）に示す様にノード“unspecified node”をマウス（入力装置250に設けられている）を操作して選択し、その後、所定の操作を行って、図6（c）に示す様にポップアップメニュー610を表示させる。そして、ポップアップメニュー610の“set string”の項目を選択して、図6（d）に示す様に文字列を記述する。続いて、ポップアップメニュー610の“make child”の項目を選択して、図6（e）に示す様にノード“節”の子ノードを作成する。引き続いて、ポップアップメニュー610の“make brother”の項目を選択して、図6（f）に示す様に弟ノード作成する。こうして作成された図6（f）に示す様なグラフは直接文書構造パターンとして用いられる。すなわち、図6（f）に示す内容が、パターン記述情報20であり、また文書構造パターン40でもある。

【0055】次に、再編成部50により再編成される再編成構造化文書70について説明する。

【0056】ファイルとして保管されている文書のままでは、文書内部の構造へのアクセスができずパターン処理に不利なので、ファイルを走査して構造を再編成する。ただし、一度にファイル全体を解析する必要はなく、照合部80が必要するとき、必要となっている部分のみを解析して出力するようにする。

【0057】この解析処理としては、ファイルの必要とする箇所にファイルポインタを移動して可変長レコードを切り出し、そのレコードに予め記述されている構造上での位置情報をもとに木構造（或いは部分木構造）を再構成するようになっている。再編成して得られた再編成構造化文書の一例を図7に示す。この図に示す様に文書は、階層構造（木構造）として表現される。なおこの実施例では、再編成構造化文書における矩形図形を文書ノードということにする。

【0058】次に、照合部80による照合処理について、図8に示すフローチャートを参照して説明する。

【0059】照合部80は、current-nodeを、再編成構造化文書の構造における最初の文書ノードにし（ステップ801）、current-pat を、文書構造パターンの構造における最初の単純パターンにする（ステップ802）。

【0060】その文書ノードおよび単純パターンは指定された接続条件に一致するか否かを判断し（ステップ803）、一致する場合は、current-pat とcurrent-nodeとが一致するか否かを判断する（ステップ804）。

【0061】ステップ804において一致する場合は、current-nodeを次の文書ノードにし（ステップ805）、その後、単純パターンが終りか否かを判断する（ステップ806）。

【0062】ステップ806において単純パターンが終了した場合は、その旨が照合部80から出力処理部90に通知される。出力処理部90では、その一致した文書ノードを出力する（ステップ807）。このようにして出力される文書ノードは、ディスプレイ240に表示されたり、あるいはファイルとして保管される。さらには、その文書ノードを他の（或いは同一の）文書中の特定領域に流し込むことによって、文書内容を編集することができる。このように検索して一致した文書ノードすなわち文書構成要素を、削除や置換したり、他の文書に挿入することができる。

【0063】ところでステップ807が終了すると、照合部80は、current-pat を前の単純パターンにし（ステップ808）、その後、文書ノードが終りか否かを判断す（ステップ809）。

【0064】文書ノードが終了したら処理を終了し、まだ文書ノードがある場合は上記ステップ803に戻りこのステップ以降を実行する。

【0065】なお、ステップ806において単純パターンが終りの場合はステップ809に進む。

【0066】また上記ステップ803において指定された接続条件に不一致の場合、ステップ804において一致しない場合は、current-pat は最初の単純パターンか否かを判断し（ステップ810）、単純パターンであれば、一致していた文書ノードまで戻り（ステップ811）、その後、ステップ805に進む。

【0067】ステップ810において単純パターンでない場合はcurrent-pat を前の単純パターンにし（ステップ812）、その後、ステップ811に進む。

【0068】以上のような処理を行って得られた照合結果を図9に示す。この図に示した例では、図3に示した文書構造パターンと図7に示した再編成構造化文書とを

照合した場合の結果を示している。

【0069】この例においては、図9中点線で示されるように、単純パターン901Aと文書ノード901Bとが一致し、単純パターン902Aと文書ノード902Bとが一致し、単純パターン903Aと文書ノード903Bとが一致し、単純パターン904Aと文書ノード904Bとが一致し、単純パターン905Aと文書ノード905Bとが一致し、単純パターン906Aと文書ノード906Bとが一致している。

【0070】この図から分かるように、文書構造パターン（階層構造情報）が分かれば、例えば、単純パターン906Aの文字列が分からない場合であっても、照合処理することにより、その単純パターン906Aに対応する文書ノード906Bを得ることができ、その結果として、表題は「魔神の宅配便」であるということが分かる。

【0071】また単純パータン903Aの文字列「introduction」に一致するところが、文書ノード903B、907に存在していた場合であっても、必ず表題になっているもののみしか一致しないので（この例では文書ノード903Bのみ一致）、確実に検索することができる。

【0072】以上説明したように第1の実施例によれば、文字列パターンに加えて、適切な方法で文書の構成要素の接続関係を示すことによって、誤りなく必要とするものが得られる。

【0073】次に第2の実施例を図10乃至図15を参照して説明する。

【0074】図10は、本発明に係る文書処理装置の第2の実施例を示す機能ブロック図である。この機能ブロック図は、図1に示した第1の実施例の機能ブロック図の構成において、適合ノード蓄積部1010、命令処理処理部1020を追加し、出力処理部90を出力処理部1030に変更した構成になっている。なお図10において、図1に示した構成要素と同様の機能を果たす部分には同一の符号を付している。

【0075】適合ノード蓄積部1010は、照合部80の照合結果である文書構成要素を蓄積する。

【0076】命令処理部1020は、コマンドラインあるいは標準入力から与えられる所定の処理を解釈し、この解釈結果を出力処理部1030に与える。なお所定の処理には、“挿入する”、“置換する”、“削除する”の処理が含まれている。

【0077】出力処理部1030は、適合ノード蓄積部1010に蓄積されている文書構成要素に対して、命令処理部1020からの処理命令に従って処理を実行し、出力する。この出力は、次の処理のための標準出力に出力しても良い。

【0078】なお上記標準入力及び標準出力とは、UNIX（オペレーティングシステム）における標準入力及び標準出力のことである。

【0079】この図10に示した装置も、図2に示した第1の実施例のハードウェア構成で実現することができる。ここで図10に示した構成要素と図2に示した構成要素との対応関係について説明する。図10に示した適合ノード蓄積部1010は図2に示した主メモリ220に対応し、図10に示した命令処理部1020及び出力処理部1030は共に図2に示したCPU210に対応している。他の構成要素については第1の実施例と同様である。

【0080】この第2の実施例は、基本的には第1の実施例と同様である。第1の実施例と異なるのは、構造化文書中から、文書構造パターンに一致する構造を抽出し、この抽出した構造に対して、“挿入”、“置換”、“削除”などの処理を施すという点である。

【0081】そこで、第2の実施例における文書編集処理について、図11乃至図15を参照して説明する。

【0082】パターン記述情報20として、

注　　…（2）

（節／本文段落／スタミナX）＃＇

ここで、／は包含関係を示す記号

＃は順序関係を示す記号

＇は処理対象となるノードを示す記号

が記述されメモリ10に記憶されている。

【0083】次に解釈部30は、図4に示した第1の実施例のパターン解釈処理手順と同様の処理を実行して、上記（2）に示すパターン記述情報20から、図11に示す様な文書構造パターンを生成し、これを文書構造パターン40としてメモリ10に格納する。このとき処理対象となるノードは「注」であると解釈する。図11では、処理対象となるノードの目印として二重枠で囲んで表記している。この場合も、図5に示した第1の実施例の具体例の様にパターン解釈されていく。

【0084】この第2の実施例でも、図形エディタを用いて描画したグラフ（グラフ理論におけるグラフ）を解釈して文書構造パターンを認識することができる。図形エディタを用いたパターン記述の方法は、図6を用いて説明した第1の実施例と同様である。

【0085】一方、再編成部50による再編成処理結果は、図12に示す内容であり、メモリ10に再編成構造化文書70として格納される。

【0086】そして照合部80が、図8に示した第1の実施例の照合処理手順と同様の処理を実行して、図11に示す文書構造パターンと、図12に示す再編成構造化文書とを照合する。この結果として、図13に示すような照合結果が得られることとなる。図13においては、単純パターン1310と文書ノード1310A、1310Bとが一致し、単純パターン1320と文書ノード1320A、1320Bとが一致し、単純パターン1330と文書ノード1330A、1330Bとが一致し、単

純パターン１３４０と文書ノード１３４０Ａ、１３４０Ｂとが一致しいる。このとき照合部８０は、パターン記述情報中で指示されている処理対象ノードに対応する構造化文書中のノードを適合ノード蓄積部１０１０に格納する。この例では「注」のノード１３４０Ａ、１３４０Ｂが蓄積される。なおノードの全情報を格納する必要はなく、ノードへのリンクを記憶するようにしても良い。

【００８７】命令処理部１０２０に、例えば「insert “スタミナXは絶倫製薬の登録商標です” as lastChild」というコマンドラインが与えられると、命令処理部１０２０は、そのコマンドラインを、出力処理部１０３０の動作を決定する変数として解釈し、この解釈結果に応じた処理を実行する。この処理を図１４を用いて説明する。

【００８８】最初にinsertを解釈して、“挿入する”を示すinsertのコマンドを出力処理部１０３０に設定する。コマンドの設定は、解釈したコマンドを設定するようにしても良いし、出力処理部１０３０に予め設定されたコマンド群を用意しておき、解釈したコマンドに該当するものを設定するようにしても良い。ここでは、後者の方法を採用しており、“挿入する”、“置換する”、“削除する”にそれぞれ対応するコマンドinsert, replace, delete が予め用意されている。これらのコマンド以外にも、insert from file"tottekoi", insert from stdin, remove などのコマンドを用意することもできる。

【００８９】次に“スタミナXは絶倫製薬の登録商標です”を解釈して、処理上必要となるデータ領域（ここではbufferという名前で参照される）に文字列を出力処理部１０３０に複写する。

【００９０】最後にas lastChildを解釈して、末子であるということを示すフラグ１を出力処理部１０３０に設定する。

【００９１】出力処理部１０３０では、「insertのコマンド」、「buffer＝スタミナXは絶倫製薬の登録商標です」、「フラグ１」が設定されると、これらの情報に基づいて、適合ノード蓄積部１０１０に蓄積されているノードに対して処理を施し、この結果をファイルストリームに出力する。この出力結果を図１５に示す。この図１５に示す例では、注のノード１３４０Ａ、１３４０Ｂの子供として“スタミナXは絶倫製薬の登録商標です”のノード１５００Ａ、１５００Ｂが接続されている。なお処理と出力は同時に実行することができる。更には対象ノードの照合の検証とも同時に実行することができる。

【００９２】以上説明したように第２の実施例によれば、構造化文書内のオブジェクト間の関係を利用したパターン、つまり基準となる階層構造のパターンとのパターンマッチングを行うようにしているので、構造化文書における正確な情報（文書構成要素）にアクセスすることができると共に、階層上のデータ（文書構成要素）の位置を簡単に指定することができる。

【００９３】また１つのパターンを使用して上述したようなパターンマッチングを行うようにしているので、構造化文書中の複数の書き換え又は挿入位置を指定することができる。

【００９４】更にユーザとのインタラクションなしで正確なアクセスを実行することができるので、構造を持った文書のバッチ処理が可能となる。

【００９５】次に第３の実施例について、図１６乃至図２０を参照して説明する。

【００９６】図１６は、本発明に係る文書処理装置の第３の実施例を示す機能ブロック図である。この機能ブロック図は、図１０に示した第２の実施例の機能ブロック図の構成において、適合ノード蓄積部１０１０を削除し、削除情報蓄積部１０４０を追加した構成になっている。

【００９７】削除情報蓄積部１０４０は、照合部８０の照合により一致した文書ノードと、当該文書ノードの親である文書ノードとを対応付けして蓄積（この蓄積の構造ついては後述する）し管理する。

【００９８】出力処理部１０３０は、命令処理部１０２０から“削除する処理”が渡されると、削除情報蓄積部１０４０に蓄積されている照合部８０の照合により一致した文書ノードの親である文書ノードから、削除情報蓄積部１０４０に蓄積されている照合部８０の照合により一致した文書ノードに関する情報（例えば文書ノード、そのノードの位置情報）を取り除くと共に、当該親の文書ノードから削除されない子供の文書ノードを出力する。

【００９９】なお構造化文書においては、一般的に、各ノード間の関係を表現する際に、親ノードには自己の子供である子ノードを示す情報が含まれており、一方、子ノードに自己の親である親ノードを示す情報が含まれているので、出力処理部１０３０は、“削除する処理”を受け取ったときは、親の文書ノードから、削除すべき文書ノードを示す情報を削除するようにしている。このとき、削除される文書ノードに含まれている親の文書ノードを示す情報は削除してもしなくとも良い。但し、親の文書ノードから削除すべき文書ノードを示す情報を削除することにより、当該削除すべき文書ノードは、親の文書ノードとの関連性がなくなり出力されないので、処理効率の点からいって、削除される文書ノードに含まれている親の文書ノードを示す情報は削除しない方が良い。

【０１００】図１７は、削除情報蓄積部１０４０に蓄積される削除情報の構造の一例を示している。この実施例では、削除情報の構造を、削除される文書ノード（以下、子ノードという）のリスト（以下、子リストという）を持った、その親の文書ノード（以下、親ノードという）のリスト（以下、親リストという）として表現するようにしている。因みに図１７においては、親リスト

には、２つの子ノードＣ1-1、Ｃ1-2の子リストを持つ親ノードＰ１と、２つの子ノードＣ2-1、Ｃ2-2の子リストを持つ親ノードＰ２とが登録されている。

【０１０１】なお、上記の例では削除される子ノードを登録するようにしているが、削除される子ノードの情報としては、何番目の子供が削除されるべきか、という情報で十分である。

【０１０２】次に、削除情報蓄積部１０４０による削除情報の作成処理について、図１８に示すフローチャートを参照して説明する。

【０１０３】削除情報蓄積部１０４０は、初期化として親リストを空にし（ステップ１０５１）、次に、照合部８０から渡される照合結果つまりパターンと一致するノード（以下、これをノードＣとする）を順次受け取ると共に、当該ノードＣを１つ取り込む（ステップ１０５２）。

【０１０４】次に削除情報蓄積部１０４０は、取り込んだノードＣの親ノードＰは親リストに未登録か否かを判断する（ステップ１０５３）。

【０１０５】ステップ１０５３において親ノードＰは登録済みの場合は、ノードＣは親ノードＰの子リストに未登録か否かを判断する（ステップ１０５４）。

【０１０６】ステップ１０５４においてノードＣは登録済みの場合は、照合部８０からの次のノードＣを取り込む（ステップ１０５５）。

【０１０７】なお、ステップ１０５４においてノードＣが未登録の場合は、ノードＣを親ノードＰの子リストに新規登録し（ステップ１０５６）、その後、ステップ１０５５に進み、またステップ１０５３において親ノードＰが未登録の場合は、親リストに新規登録し（ステップ１０５７）、その後、ステップ１０５４に進む。

【０１０８】ところで上記ステップ１０５５を終了した場合、削除情報蓄積部１０４０は、照合部８０から渡されるノードは終りか否かを判断し（ステップ１０５８）、ノードが終りの場合には処理を終了し、一方、まだノードが存在している場合は、上記ステップ１０５３に戻り、照合部８０から渡されるノードが終了するまで、ステップ１０５３～１０５８を繰り返す。

【０１０９】ここで、図１７に示した例を用いて、削除情報の作成処理を説明する。

【０１１０】図１７において、親ノードＰ１が親ノードＰ２より早く出現するものとし、今現在、親リストは空き状態とする。

【０１１１】このような状態で、上記ステップ１０５３において、子ノードＣ1-1が取り込まれた後、ステップ１０５３が実行された際には、子ノードＣ1-1の親ノードＰ１は未登録であるので、この場合はステップ１０５７に進み、このステップにより親ノードＰ１が親リストに登録される。このステップ１０５７終了後はステップ１０５４に移行するが、子ノードＣ1-1は親ノードＰ１の子リストには登録されていないので、ステップ１０５６に進み、このステップにより子ノードＣ1-1が親ノードＰ１の子リストに登録される。

【０１１２】そしてステップ１０５５、１０５８が実行されることとなり、この場合は、照合部８０からのノード（つまり照合結果であるノード）がまだ存在するので、上記ステップ１０５３に戻り、このステップにより子ノードＣ1-2についての処理が実行される。この場合は、ステップ１０５３においては「ＮＯ」（つまり親ノードＰ１は親リストに登録済み）となるので、ステップ１０５４に進み、このステップにおいては「ＹＥＳ」（子ノードＣ1-2は親ノードＰ１の親リストには未登録）なので、ステップ１０５６に進み、このステップにより子ノードＣ1-2が親ノードＰ１の親リストに登録される。

【０１１３】以下同様に、子ノードＣ2-1、Ｃ2-2についての処理が行われる。

【０１１４】なお図１７において、削除すべきノードが、子ノードＣ1-1、Ｃ1-2、親ノードＰ１であった場合は、親リストに、例えば親ノードＰ１が削除される旨の情報のみを登録し、子ノードＣ1-1、Ｃ1-2については登録しないようにする。何故ならば、削除されるノードから更に削除されるということは無いので、親ノードＰ１が削除される旨のみを登録すれば良いこととなる。従って、子ノードＣ1-1、Ｃ1-2のリストへの登録を行う必要がないので、処理効率及び記憶使用効率を向上させることができる。

【０１１５】次に、出力処理部１０３の出力処理について、図１９に示すフローチャートを参照して説明する。

【０１１６】出力処理部１０３は、命令処理部１０２０からの“削除する処理”を受け取ると（ステップ１０６１）、処理の対象をファイル先頭に移動し（ステップ１０６２）、その後、そのファイルから、文書のルートノード（これはファイル先頭）であるノードＮを１つ読み取り（ステップ１０６３）、該ノードＮについて、出力すべきノードを出力するノード出力処理を実行する（ステップ１０６４）。すなわちステップ１０６４においては、ノードＮの下位に存在するノードが出力されることになる。

【０１１７】このステップ１０６４のノード出力処理について、図２０に示すサブルーチンを参照して説明する。

【０１１８】出力処理部１０３は、ノードＮは親リストに登録されているか否かを判断する（ステップ１０７１）。ここで、ノードＮはルートノードであるので、このノードＮが、親リストに登録されているということは、必ず削除されるノード（すなわち子ノード）が存在することを意味しており、一方、親リストに登録されていない場合は、削除されるノードが存在しないことを意味している。

【0119】ところで、ステップ1071において登録済みの場合は、ノードNについて変更を施して出力する（ステップ1072）。このステップ1072においては、削除情報蓄積部1040に蓄積されている親リストつまり削除情報に基づいて、ノードNについて、親リストに登録されている親ノードから、該親ノードの子リストに登録されている子ノードに関する情報（例えば子ノードを示す情報）を取り除く。この処理が終了した後は、ノードNには出力すべきノードのみが存在していることになる。

【0120】そしてステップ1072を終了した後、出力処理部103は、ノードNについての子リストに含まれない子をノード（つまり出力すべきロノード）に対して、再帰的にノード出力処理を実行する（ステップ1073）。

【0121】一方、ステップ1071において未登録の場合は、ノードNをそのまま出力し（ステップ1074）、そのノードNの子ノード全てに対して、再帰的にノード出力処理を実行する（ステップ1075）。

【0122】以上説明したように第3の実施例によれば、パターン照合して削除されるノード（ノードC）が検出されると、このノードCと該ノードCの親のノード（ノードP）とを対応して記憶し、そして、親のノードPからノードCに関する情報（ノードCを示す情報）を取り除いて、その親のノードPを出力するようにしているので、ユーザが、親のノードPの内容を変更する必要がない。このため構造化文書に対する編集処理の作業効率を向上させることができる。

【0123】次に第4の実施例を図21乃至図27を参照して説明する。

【0124】図21は本発明に係る文書処理装置の第4の実施例を示す機能ブロック図である。同図において、文書処理装置は、メモリ1610、解釈部1620、再編成部1630、ソース文書ファイル群1640、ターゲット文書ファイル群1650、照合部1660、複数抽出点蓄積部1670、複数挿入点蓄積部1680、出力処理部1690、抽出挿入関係ルール指定部1700を備えている。

【0125】メモリ1610には、第1の実施例で説明したようなパターン記述情報1611が記憶されており、解釈部1620は、メモリ1610からパターン記述情報1611を読み出して解釈し、この解釈結果を第1の実施例で説明したような文書構造パターン1612としてメモリ1610に記憶する。

【0126】再編成部1630は、ソース文書ファイル群1640に保持されている複数の構造化文書内を走査して、これらの構造化文書を照合処理可能な形式の構造に再編成し、この結果をソース再編成構造化文書群1613としてメモリ1610に格納する。同様にして、ターゲット文書ファイル群1650内の複数の構造化文書についても、ソース再編成構造化文書群1614としてメモリ1610に格納する。

【0127】この第4の実施例において、構造化文書とは、章、節といった文書構造と文書内容とを一緒に持つフォーマットによる文書表現を意味している。1つのファイル内に複数の構造化文書を含むとは、図22に示す様に、1つのファイル内に、異なる論理根を持つ論理構造が複数含まれることを示している。論理根が異なる論理構造は互いに独立であり、且つ部分構造が共有されることはない。部分構造とは、構造化文書における一部分の構造のとであり、例えば図22中点線で囲まれた構造化文書においては、「節」というノード以下の構造や、「表題」というノード以下の構造などである。

【0128】ソース文書ファイルとは、部分構造を他の文書へ挿入するために、文書構造パターンに適合する部分構造が抽出される文書ファイル（つまりソース側の文書ファイル）のことである。結果としてこの文書ファイルの内容は変更されることはない。

【0129】一方、ターゲット文書ファイルとは、他の文書からの部分構造を挿入するために、文書構造パターンに適合する部分構造が抽出される文書ファイル（つまりターゲット側の文書ファイル）のことである。結果としてこの文書ファイルの内容は変更される。

【0130】またソース再編成構造化文書とは、ソース文書ファイル内の構造化文書に対する再編成処理の結果である再編成構造化文書のことである。

【0131】一方、ソース再編成構造化文書とは、ターゲット文書ファイル内の構造化文書に対する再編成処理の結果である再編成構造化文書のことである。

【0132】照合部1660は、複数ファイル抽出点認識部1661と、複数ファイル挿入点認識部1662とを有している。複数ファイル抽出点認識部1661は、ソース構造化文書群をそれぞれ格納する複数のファイルに対して、文書構造パターンによる照合によりファイルを走査し、複数の部分構造の抽出点を認識し、この認識結果を複数抽出点蓄積部1670に格納する。このときファイル名と抽出点の対の情報を格納する。一方、複数ファイル挿入点認識部1662は、ターゲット構造化文書群をそれぞれ格納する複数ファイルに対し、文書構造パターンによりファイルを走査し、複数の部分構造に対する挿入点を認識すると共に、この認識結果を複数挿入点蓄積部1680に格納する。このときファイル名と挿入点の対の情報を格納する。なお複数抽出点蓄積部1670と複数挿入点蓄積部1670とは独立しているが、抽出点の情報と挿入点の情報とを区別するようにして、これらの情報を1つの蓄積部に蓄積するようにしても良い。

【0133】抽出挿入関係ルール指定部1700は、抽出点と挿入点との対応関係を、ファイルを跨がる1対1、あるいはファイルを跨がる複数n対1のいずれかの

ルールを出力処理部１６９０に与える。

【０１３４】出力処理部１６９０には、複数ファイル部分構造抽出挿入部１６９１が設けられており、複数ファイル部分構造抽出挿入部１６６１は、抽出挿入関係ルール指定部１７００から与えられた抽出点と挿入点との対応関係の情報に基づいて、複数抽出点蓄積部１６７０に蓄積されている抽出点に対応する部分構造から、複数挿入点蓄積部１６８０に蓄積されている挿入点に対応する部分構造分への文書構造の挿入操作を実行する。

【０１３５】この図２１に示した装置も、図２に示した第１の実施例のハードウエア構成で実現することができる。ここで、図２１に示した機能ブロック図の構成要素と図２に示したブロック図の構成要素との対応関係について説明する。図２６に示したメモリ１６１０は図２に示した主メモリ２０に対応し、図２１に示した解釈部１

（節／表題／まとめ）＃本文段落 …（３）

ここで、／は包含関係を示す記号
＃は順序関係を示す記号
が記述されメモリ１６１０に記憶されている。

【０１３９】次に解釈部１６３０によって、図４に示す第１の実施例のパターン解釈処理手順が実行されることにより上記（３）のパターン記述情報が解釈され、更にこの結果が文書構造パターン１６１２としてメモリ１６１０に記憶される。

【０１４０】続いて再編成部１９３０によって、ソース文書ファイル群１６４０とターゲット文書ファイル群１６５０とが再編成され、更にこれらの結果が、ソース再編成構造化文書群１６１３、ターゲット再編成構造化文書群１６１４としてメモリ１６１０に記憶される。

【０１４１】続いて照合部１６６０の複数ファイル抽出点認識部１６６１による抽出点認識処理について、図２３を参照して説明する。図２３はその処理動作を示すフローチャートである。

【０１４２】複数ファイル抽出点認識部１６６１は、最初のソース文書ファイル（ソース再編成構造化文書群１６１３中の１つのファイル）をメモリ１６１０から読み込んで（ステップ１８０１）、ソース文書ファイルは終りか否かを判断し（ステップ１８０２）、終りの場合には処理を終了し、一方、終りでない場合は、ファイル内の全ての構造化文書（つまり論理根を持つ文書）に対する処理が終了したか否かを判断する（ステップ１８０３）。

【０１４３】ここで、まだ未処理の構造化文書が存在している場合は、その構造化文書に対するパターン照合処理を実行し（ステップ１８０４）、その照合処理結果である抽出点を複数抽出点蓄積部１６７０に蓄積する（ステップ１８０５）。

【０１４４】上記ステップ１８０３において、全ての構造化文書について処理した場合は、次のソース文書ファイルをメモリ１６１０から読み込み、その後、上記ステップ１８０２に戻る。

【０１４５】なおステップ１８０４のパターン照合処理は、図８に示す第１の実施例の処理手順と同様である。

【０１４６】同様にして、複数ファイル挿入点認識部１６６２は、ターゲット文書ファイル（ターゲット再編成構造化文書群１６１４）に対する挿入点の認識処理を実行する。この結果は、複数挿入点蓄積部１６８０に蓄積される。

【０１４７】すなわち、複数ファイル抽出点認識部１６６１と複数ファイル挿入点認識部１６６２は基本的には同様の処理を実行し、異なるのは、対象となる文書ファイル（構造化文書）がソースであるかターゲットであるかという点である。

【０１４８】ここで、抽出点の認識処理結果の様子を図２４に示す。図２４において、ファイル１、ファイル２は、ソース再編成構造化文書を示しており、またハッチングの掛った部分が、文書構造パターン１６１２に適合した部分である。この図２４から分かるように、ファイル内の複数の構造化文書及び複数のファイルに跨がって、構造がパターンマッチングされ適合されている。この例での抽出点は、ハッチングの掛った部分の「節」というノードの直前の位置（つまり「論理根」というノードとの接続点の位置）である。この抽出点は、各ファイル毎に抽出点の列として複数抽出点蓄積部１６７０に蓄積される。

【０１４９】同様に挿入点の認識処理結果も、図２４に示す様に、文書構造パターン１６１２に適合した部分が認識されることとなる。挿入点についても上記同様に考えることができる。

【０１５０】以上の説明から分かるように、この第４の実施例においては、図２４に示すように、文書構造パターン１６１２に適合する部分構造（ハッチング部分）を抽出することが、本来の目的ではなく、「節」というノードを抽出することが目的なのである。しかし、図２４

６２０、再編成部１６３０、照合部１６６０、出力処理部１６９０及び抽出挿入関係ルール指定部１７００は共に図２に示したＣＰＵ２１０に対応し、ソース文書ファイル群１６４０及びターゲット文書ファイル群１６５０は共に図２に示したディスク２３０に対応している。

【０１３６】この第４の実施例も、基本的には第１の実施例と同様である。第１の実施例と異なるのは、１つのファイル内の複数の構造化文書に対して、文書構造パターンに一致する構造を抽出する点である。また複数の構造化文書を有するファイルを複数設け、これらのファイル内の複数の構造化文書に対して照合する点も異なっている。

【０１３７】そこで、第４の実施例における文書編集処理について、図２３乃至図２７を参照して説明する。

【０１３８】パターン記述情報２０として、

に示されるように、「節」というノード以下の構造には各種の部分構造が接続されているので、所望の「節」というノードを抽出するために、文書構造パターン１６１２との照合を実施しているのである。

【０１５１】次に、出力処理部１６９０の複数ファイル部分構造抽出挿入部１６９１の出力処理について、図２５を参照して説明する。図２５はその処理動作を示すフローチャートである。

【０１５２】複数ファイル部分構造抽出挿入部１６９１は、複数抽出点１蓄積部１６７０から各ファイル毎の抽出点の列を得る。これらをＡ［ｉ］＝（file名、抽出点）に順に格納すると共に（ステップ２００１）、複数挿入点蓄積部１６８０から各ファイル毎の挿入点の列を得る。これらをＢ［ｊ］＝（file名、挿入点）に順に格納する（ステップ２００２）。

【０１５３】次に、挿入抽出関係ルール指定部１７００から指定された抽出点と挿入点との対応関係のルールが“ファイルを跨がる１対1”であるか否かを判断する（ステップ２００３）。

【０１５４】ここで、“ファイルを跨がる１対１”の場合は、ｉ＝１、ｊ＝１と定義し（ステップ２００４）、Ａ［ｉ］あるいはＢ［ｊ］が終りか否かを判断する（ステップ２００５）。

【０１５５】ここで、終りでない場合は、Ａ［ｉ］に示される抽出点に基づいて、ソース文書ファイル群から部分構造を抽出すると共に（ステップ２００６）、この部分構造をＢ［ｊ］に示される挿入点に挿入する（ステップ２００７）。

【０１５６】その後、ｉ＝ｉ＋１、ｊ＝ｊ＋１と再定義した後（ステップ２００８）、上記ステップ２００５に戻る。ステップ２００５においてＡ［ｉ］あるいはＢ［ｊ］が終りの場合は、結果を出力する（ステップ２００９）。

【０１５７】上記ステップ２００３においてルールが“ファイルを跨がる１対1”でない場合は、ルールが“ファイルを跨がる複数ｎ対１”であるか否かを判断する（ステップ２０１０）。そうであれば、ｊ＝１と定義し（ステップ２０１１）、その後、Ｂ［ｊ］が終りであるか否かを判断する（ステップ２０１２）。

【０１５８】ここで、終りの場合は、Ａ［１］～Ａ［ｎ］に示される抽出点に基づいて、ソース文書ファイル群から部分構造を全て抽出し、これらＡ［１］～Ａ［ｎ］までの部分構造を兄弟として繋ぐと共に（ステップ２０１３）、兄弟として繋がれた構造を、Ｂ［ｊ］に示される挿入点に挿入する（ステップ２０１４）。この挿入点に対して、兄、弟、子供として挿入することができる。

【０１５９】上記ステップ２０１４を終了した後はｊ＝ｊ＋１と再定義し（ステップ２０１５）、その後、上記ステップ２０１２に戻る。すなわち結果として、Ｂ［１］～Ｂ［ｎ］の各挿入点に、兄弟として繋がれたＡ［１］～Ａ［ｎ］までの部分構造が挿入される。

【０１６０】上記ステップ２０１２においてＢ［ｊ］が終了した場合は上記ステップ２００９に進む。

【０１６１】上記ステップ２０１０においてルールが“ファイルを跨がる複数ｎ対１”でない場合は挿入処理は行わない（ステップ２０１６）。

【０１６２】なお複数ファイル部分構造抽出挿入部１６９１は、ソース文書ファイル群の抽出点、ターゲット文書ファイル群の挿入点のいずれかの数が多いときは挿入処理を行わず、ステータスを返す。

【０１６３】例えば、抽出点の数＞挿入点の数、のときステータスの値が１
抽出点の数＜挿入点の数、のときステータスの値が２
この結果として、出力処理部１６９０からは、エラー通知が出力される。

【０１６４】また、抽出点の数＝挿入点の数、のときは０のステータスを返す。この結果として、出力処理部１６９０からは、変更後のターゲット文書ファイルが出力されることとなる。

【０１６５】ここで、ファイルを跨がる１対１のルールに基づく挿入結果の様子を図２１に示し、またファイルを跨がる複数ｎ対１のルールに基づく挿入結果の様子を図２２に示す。

【０１６６】以上説明したように第４の実施例によれば、複数の文書ファイルであって、且つ１つのファイル中に複数の構造化文書文書が保存されている場合であっても、構造化文書内のオブジェクト間の関係を利用したパターン、つまり基準となる階層構造のパターンとのパターンマッチングを行うようにしているので、構造化文書における正確な情報（文書構成要素）にアクセスすることができると共に、階層上のデータ（文書構成要素）の位置を簡単に指定することができる。

【０１６７】すなわち、ターゲット側の複数のファイルそれぞれに保存されている複数の構造化文書から抽出された文書構成要素に対する、ソース側の複数のファイルそれぞに保存されている複数の構造化文書から抽出された文書構成要素の挿入操作を一度に自動的に行うことができるということである。

【０１６８】次に第５の実施例を図２８乃至図３１を参照して説明する。

【０１６９】図２８は本発明に係る文書処理装置の第５の実施例を示す機能ブロック図である。この機能ブロック図は、図１に示した第１の実施例の機能ブロック図の構成において、ファイル位置情報保持部２３１０、属性情報指定部２３２０を追加し、出力処理部９０を出力処理部２３３０に変更した構成になっている。なお図２３において、図１に示した構成要素と同様の機能を果たす部分には同一の符号を付している。

【０１７０】ファイル位置情報保持部２３１０は、照合

部８０の照合結果である文書構成要素のファイル内の位置情報を保持する。

【０１７１】属性情報指定部２３２０は、文書構成要素の属性情報を指定するものであり、属性値の参照のときは属性名を指定し、属性値の変更のときは属性名及び属性値を指定する。

【０１７２】出力処理部２３３０は、ファイル位置情報保持部２３１０に保持されている文書構成要素のファイル内の位置情報と、属性情報指定部２３２０から指定される属性情報とに基づいて出力処理を実施する。ここで、属性値の参照のときは、該当する文書構成要素の属性名を持つ属性の属性値を出力し、一方、属性値の変更のときは、該当する文書構成要素の属性名を持つ属性を、指定された属性値に変更して出力する。

【０１７３】なおここでは構造化文書は、図２９（ａ）に示す様に各文書構成要素をノードとする木構造を持っているが、ファイル上では、図２９（ｂ）に示す様に、決められた規則に従って各文書構成要素は１列に並んでいる。また文書構成要素内の属性名と属性値の対も１列に並んでいるとする。

【０１７４】図２８に示した装置も、図２に示した第１の実施例のハードウエア構成で実現することができる。ここで図２８に示した機能ブロック図の構成要素と図２に示したブロック図の構成要素との対応関係について説明する。図２８に示したファイル位置情報保持部２３１０は図２に示した主メモリ２０に対応し、図２８に示した属性指定部２３２０及び出力処理部２３３０は共に図２に示したＣＰＵ２１０に対応している。他の構成要素については第１の実施例と同様である。

【０１７５】この第５の実施例も、基本的には第１の実施例と同様である。第１の実施例と異なるのは、構造化文書中から、文書構造パターンに一致する構造を抽出し、この抽出した構造に対して、属性の参照又は変更の処理を施すという点である。

【０１７６】なおこの第５の実施例においては、照合部８０による文書構造パターンと再編成構造化文書との照合処理までは、第１の実施例で説明した処理と同様なので、ここではその説明を省略し、属性の参照又は変更処理について説明する。

【０１７７】次に、文書処理装置の属性の参照又は変更処理について、図３０及び図３１を参照して説明する。図３０は属性の参照処理動作のフローチャートを示し、図３１は属性の変更処理動作のフローチャートを示している。

【０１７８】最初に属性の参照処理について説明する。図３０に示すように、出力処理部２３３０は、ファイル位置情報保持部２３１０から属性結果（ファイル位置）を１つ取り出し（ステップ２５０１）、文書ファイルの「読み出し位置」を読み出した照合結果に設定すると共に（ステップ２５０２）、その読み出し位置に存在する文書構成要素内から属性を１つ読み込む（ステップ２５０３）。

【０１７９】次に出力処理部２３３０は、その属性名は属性情報指定部２３２０から指定された属性名と同じであるか否かを判断する（ステップ２５０４）。

【０１８０】ここで、同一の場合はその属性値を出力用のファイルに書き出し（ステップ２５０５）、その後、属性は終りか否かを判断する（ステップ２５０６）。

【０１８１】ここで、終りでない場合は、上記ステップ２５０３に戻りこのステップ以降を実行する。すなわち１つの文書構成要素内に存在する全ての属性についての属性の参照の処理を実施する。

【０１８２】ステップ２５０６において属性が終りの場合は、照合結果は終りか、つまりファイル位置情報保持部２３１０に保持されている全ての照合結果について処理したか否かを判断する（ステップ２５０７）。

【０１８３】ここで、未処理の照合結果がある場合には上記ステップ２５０１に戻りこのステップ以降を実行し、一方、全て処理した場合は属性の参照処理を終了する。

【０１８４】なおステップ２５０４において指定された属性名でない場合は何もしないでステップ２５０６に進む。

【０１８５】次に属性の変更処理について説明する。図３１に示す様に、出力処理部２３３０は、文書ファイルの「読み出し位置」を先頭に設定すると共に（ステップ２６０１）、ファイル位置情報保持部２３１０から照合結果（ファイル位置）を１つ取り出す（ステップ２６０２）。次に、文書ファイルにおける照合結果の位置までの部分をそのまま出力用のファイルに書き出す（ステップ２６０３）。

【０１８６】続いて、文書ファイルの「読み出し位置」を照合結果に設定すると共に（ステップ２６０４）、その読み出し位置に存在する文書構成要素内から属性を１つ読み込む（ステップ２６０５）。

【０１８７】そして、その属性名は属性情報指定部２３２０から指定された属性名と同じであるか否かを判断する（ステップ２６０６）。ここで、同一の場合は属性名と指定された属性値とを出力用のファイルに書き出し（ステップ２６０７）、同一でない場合は、属性名と読み込んだ属性値とを出力用のファイルに書き出す（ステップ２６０８）。

【０１８８】ステップ２６０７あるいはステップ２６０８を終了したら、属性は終りか否かを判断する（ステップ２６０９）。ここで、終りでない場合は、上記ステップ２６０５に戻りこのステップ以降を実行する。すなわち１つの文書構成要素内に存在する全ての属性について属性の変更処理を実施する。

【０１８９】ステップ２６０９において属性が終りの場合は、照合結果は終りか、つまりファイル位置情報保持

部２３１０に保持されている全ての照合結果について処理したか否かを判断する（ステップ２６１０）。

【０１９０】ここで、未処理の照合結果がある場合には上記ステップ２６０２に戻りこのステップ以降を実行し、一方、全て処理した場合は、文書ファイルの最後までをそのまま出力用のファイルに書き出す（ステップ２６１１）。

【０１９１】この第５の実施例においては、属性の参照により取り出された属性は、外部の汎用的な演算手段を用いることにより加工することができる。その加工結果を属性情報指定部２３２０から属性情報として設定することにより、文書内に付加することができる。

【０１９２】以上説明したように第５の実施例によれば、パターン記述情報と、属性名か属性名及び属性値を設定することにより、構造化文書内のオブジェクト間の関係を利用したパターン、つまり基準となる階層構造のパターンとのパターンマッチングを実施し、マッチした部分（文書構成要素）の属性の参照又は変更操作を実行するようにしたので、属性の参照又は変更の処理を自動化することができる。またこのとき、従来の如く予めスタイルを設定しておく必要がないので、ユーザの作業量を軽減させることができる。

【０１９３】また特定部分の属性に、他の部分の属性値を元にした演算結果を設定するようなことも可能となる。

【０１９４】

【発明の効果】以上説明したように、第１の発明によれば、解釈手段によって解釈された文書構造パターンと、再編成手段によって再編成された構造化文書とを照合手段により照合し、更に出力処理手段が、その照合により一致した文書構成要素を前記構造化文書から抽出し出力するようにしているので、構造化文書に対する指定された階層構造パターンに従った文書構成要素の検索処理を行うことができるという利点がある。

【０１９５】第２の発明によれば、出力処理手段は、照合手段の照合により一致した文書構成要素に対して、指定手段により指定された所定の処理例えば削除、置換、複写などの処理を施した後、出力するようにしているので、構造化文書に対する指定された階層構造パターンに従った文書構成要素に対して、削除、置換、複写などの処理を自動的に実行することができることとなり、構造化文書のバッチ処理が可能となる。

【０１９６】第３の発明によれば、指定手段により削除処理が指定されると、出力処理手段は、蓄積手段に蓄積されている、照合手段の照合により一致した文書構成要素の親である文書構成要素から、蓄積手段に蓄積されている照合手段の照合により一致した文書構成要素に関する情報を取り除くと共に、当該親である文書構成要素からは削除されない子供の文書構成要素を出力するようにしているので、削除すべき文書構成要素の親の文書構成要素の内容を自動的に変更することができることとなり、構造化文書に対する編集操作を高率良く行うことができるという利点がある。

【０１９７】第４の発明によれば、解釈手段が、基準となる文書構成要素同志の接続関係を解釈し、また再編成手段が、格納手段に格納されているファイル内の複数の構造化文書それぞれを照合処理可能な形式の構造に再編成し、また照合手段が、解釈手段による解釈結果と、再編成手段による再編成結果とを照合し、更に出力処理手段が、照合手段の照合により一致した文書構成要素を前記再編成結果から抽出するようにしているので、複数の構造化文書から、指定された階層構造に従った文書構成要素を検索し出力することができるという利点がある。

【０１９８】第５の発明によれば、複数の構造化文書を保存した複数のファイルを対象として、文書構成要素同志の接続関係に適合する文書構成要素を再編成結果から抽出するようにしているので、複数のファイルそれぞれに複数の構造化文書が存在している場合であっても、複数のファイルおよび複数の構造化文書に跨がって、指定された階層構造に従った文書構成要素を検索し出力することができることとなり、複数の文書に対する検索処理を高速に実行することができるという利点がある。

【０１９９】第６の発明によれば、ソース側及びターゲット側それぞれの複数の構造化文書を保存した複数のファイルを対象として、文書構成要素同志の接続関係に適合する文書構成要素を再編成結果から抽出すると共に、ターゲット側の抽出結果である文書構成要素に対するソース側の抽出結果である文書構成要素の挿入を実行するようにしているので、ターゲット側の複数のファイルそれぞれに保存されている複数の構造化文書から抽出された文書構成要素に対する、ソース側の複数のファイルそれぞれに保存されている複数の構造化文書から抽出された文書構成要素の挿入を一度に行うことができる。よって文書の編集処理を迅速に行うことができるという利点がある。

【０２００】第７の発明によれば、出力処理手段は、指定された属性に関する情報に基づいて、照合手段の照合により一致した文書構成要素の属性の参照又は変更の操作を実行するようにしているので、構造化文書から、指定された階層構造に従った文書構成要素を検索し、この文書構成要素の属性に対する参照又は変更の操作を実施することができることとなり、構造化文書の文書構成要素の属性に対する操作を容易に実施することができるという利点がある。

【図面の簡単な説明】

【図１】本発明に係る文書処理装置の第１の実施例を示す機能ブロック図。

【図２】図１に示した実施例の装置を実現するためのハードウェア構成を示すブロック図。

【図３】第１の実施例における解釈部によるパターン記

述の解釈処理を説明するための図。

【図４】第１の実施例における解釈部の解釈処理動作を示すフローチャート。

【図５】第１の実施例における解釈部によるパターン記述の解釈処理過程を説明するための図。

【図６】第１の実施例における解釈部によるパターン記述の解釈処理過程を説明するための図。

【図７】第１の実施例における再編成部による構造化文書の再編成処理を説明するための図。

【図８】第１の実施例における照合部の照合処理動作を示すフローチャート。

【図９】第１の実施例における照合部による文書構造パターンと再編成構造化文書との照合処理を説明するための図。

【図１０】本発明に係る文書処理装置の第２の実施例を示す機能ブロック図。

【図１１】第２の実施例における解釈部によるパターン記述の解釈処理を説明するための図。

【図１２】第２の実施例における再編成部による構造化文書の再編成処理を説明するための図。

【図１３】第２の実施例における照合部による文書構造パターンと再編成構造化文書との照合処理を説明するための図。

【図１４】第２の実施例における命令処理部の解釈処理を説明するための図。

【図１５】第２の実施例における出力処理部の出力処理を説明するための図。

【図１６】本発明に係る文書処理装置の第３の実施例を示す機能ブロック図。

【図１７】第３実施例の削除情報の構造の一例を示す図。

【図１８】第３実施例における削除情報作成処理を示すフローチャート。

【図１９】第３実施例における出力処理部による出力処理動作を示すフローチャート。

【図２０】第３実施例における出力処理部による出力処理動作を示すサブルーチン。

【図２１】本発明に係る文書処理装置の第４の実施例を示す機能ブロック図。

【図２２】第４実施例における構造化文書を説明するための図。

【図２３】第４実施例における抽出点認識処理動作を示すフローチャート。

【図２４】第４の実施例における照合部による文書構造パターンと再編成構造化文書との照合処理を説明するための図。

【図２５】第４の実施例における出力処理部の出力処理動作を示すフローチャート。

【図２６】第４の実施例における出力処理部の出力処理を説明するための図。

【図２７】第４の実施例における出力処理部の出力処理を説明するための図。

【図２８】本発明に係る文書処理装置の第５の実施例を示す機能ブロック図。

【図２９】第５実施例における構造化文書を説明するための図。

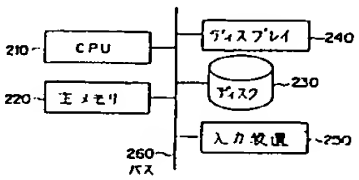
【図３０】第５実施例における文書構成要素の属性の参照処理動作を示すフローチャート。

【図３１】第５実施例における文書構成要素の属性の変更処理動作を示すフローチャート。

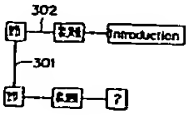
【符号の説明】

１０、１６１０…メモリ、２０、１６１１…パターン記述情報、３０、１６２０…解釈部、４０、１６１２…文書構造パターン、５０、１６３０…再編成部、６０…文書ファイル、７０…再編成構造化文書、８０、１６６０…照合部、９０、１０３０、１６９０、２３３０…出力処理部、２１０…中央処理装置、２２０…主メモリ、２３０…ディスク、２４０…ディスプレイ、２５０…入力装置、１０１０…適合ノード蓄積部、１０２０…命令処理部、１０４０…削除情報蓄積部、１６１３…ソース再編成構造化文書群、１６１４…ターゲット再編成構造化文書群、１６４０…ソース文書ファイル群、１６５０…ターゲット文書ファイル群、１６６１…複数ファイル抽出点認識部、１６６２…複数ファイル挿入点認識部、１６７０…複数抽出点蓄積部、１６８０…複数挿入点蓄積部、１６９１…複数ファイル部分構造抽出挿入部、１７００…抽出挿入関係ルール指定部、２３１０…ファイル位置情報保持部、２３２０…属性情報指定部。

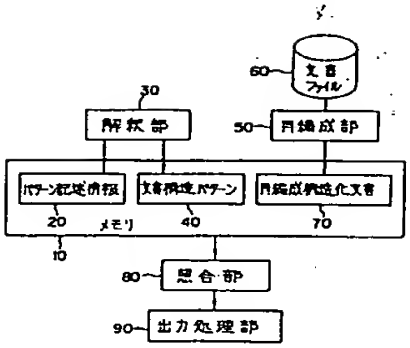
【図２】



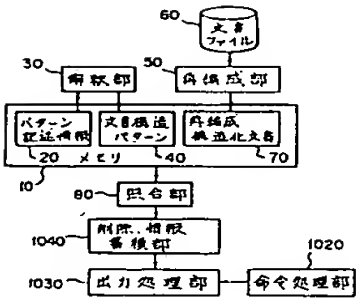
【図３】



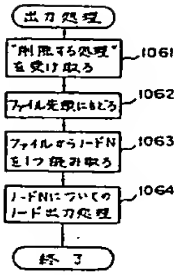
【図1】



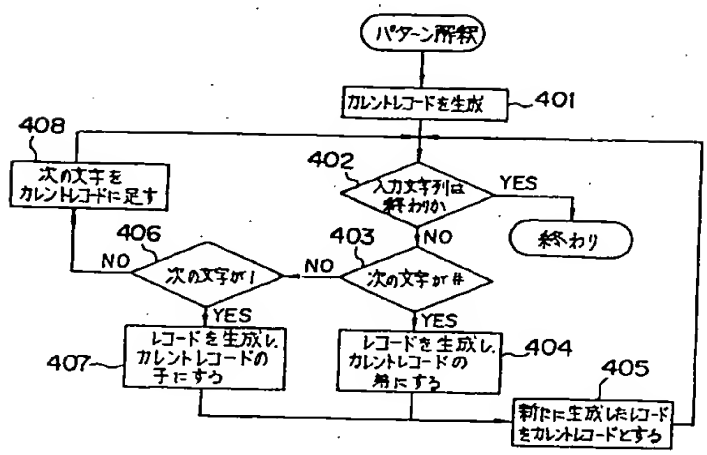
【図16】



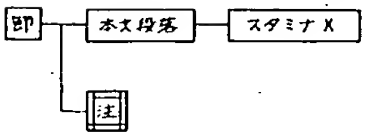
【図19】



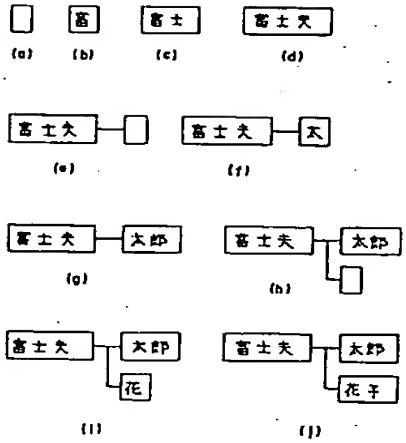
【図4】



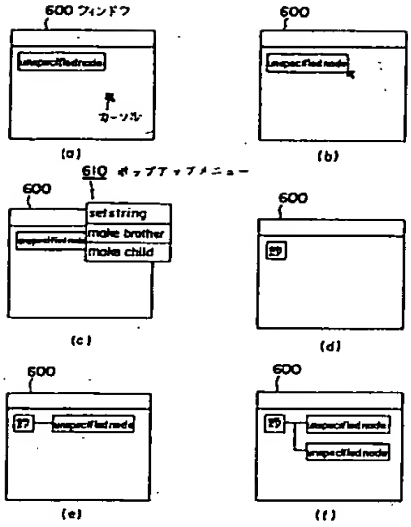
【図11】



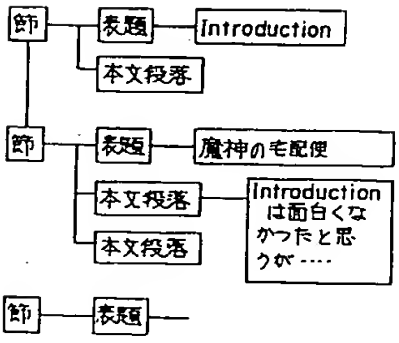
【図5】


富
富士
富士夫
太
太郎
花
花子

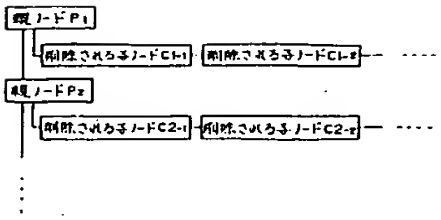
【図6】


600 ウィンドウ
600
610 ポップアップメニュー
set string
make brother
make child
600
600
600
600

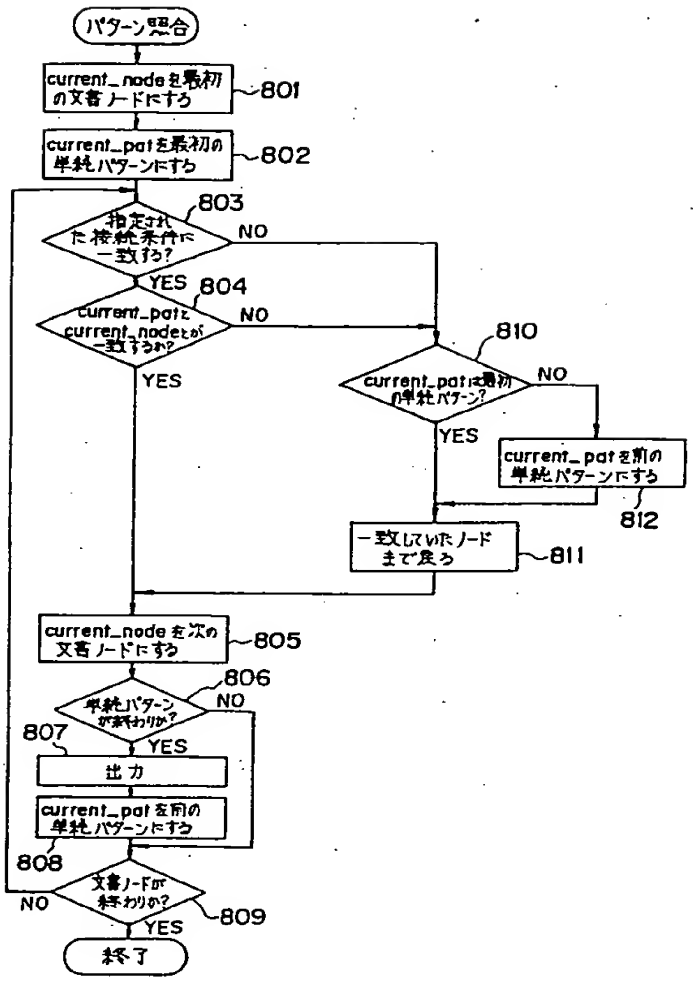
【図7】


節
表題
Introduction
本文段落
節
表題
魔神の宅配便
本文段落
Introduction は面白くなかったと思うが……
本文段落
節
表題

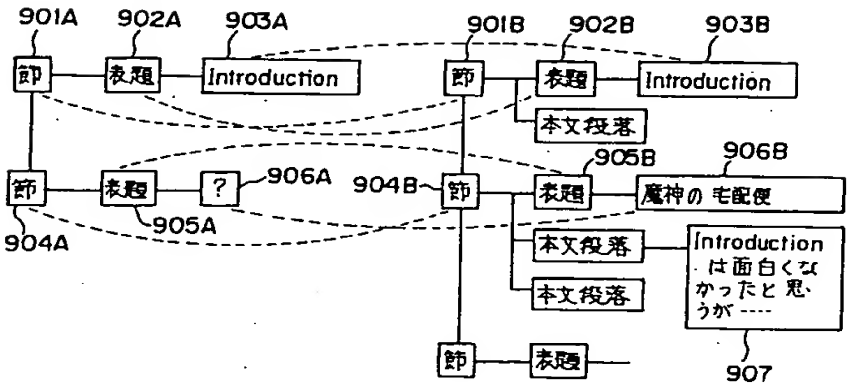
【図17】



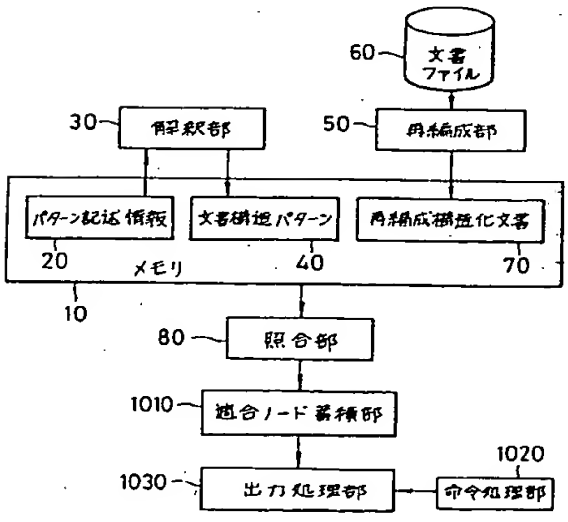
【図8】


パターン照合
current_nodeを最初の文書ノードにする
801
current_patを最初の単純パターンにする
802
803
指定された接続条件に一致する?
NO
YES
804
current_patとcurrent_nodeとが一致するか?
NO
YES
810
current_patは最初の単純パターン?
NO
YES
current_patを前の単純パターンにする
812
一致していたノードまで戻る
811
current_nodeを次の文書ノードにする
805
806
単純パターンが終わりか?
NO
807
YES
出力
current_patを前の単純パターンにする
808
文書ノードが終わりか?
NO
809
YES
終了

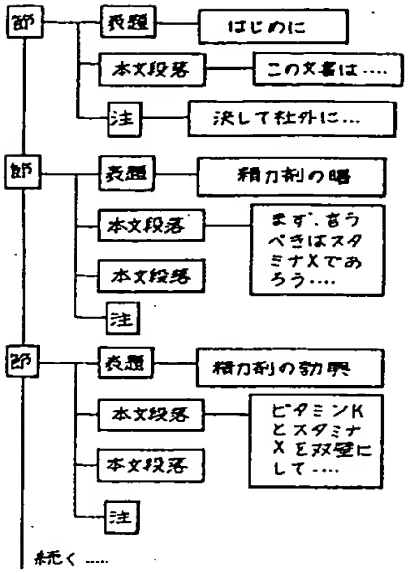
【図9】


901A
902A
903A
901B
902B
903B
節
表題
Introduction
節
表題
Introduction
本文段落
905B
906B
節
表題
?
906A
904B
節
表題
魔神の宅配便
905A
904A
本文段落
Introduction
は面白くな
かったと思
うが----
本文段落
907
節
表題

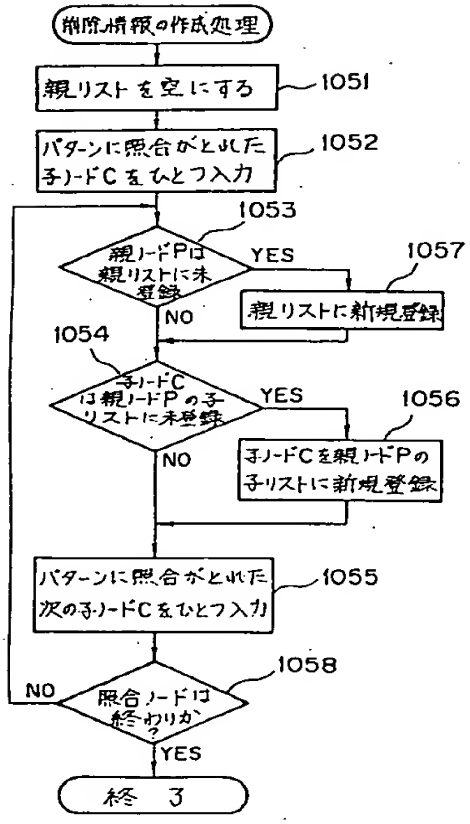
【図10】


60
文書
ファイル
30
解釈部
50
再構成部
パターン記述情報
文書構造パターン
再構成構造化文書
20
メモリ
40
70
10
80
照合部
1010
照合ノード蓄積部
1020
1030
出力処理部
命令処理部

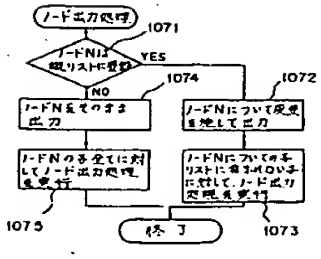
【図12】


節
表題
はじめに
本文段落
この文書は....
注
決して社外に...
節
表題
精力剤の暗
本文段落
まず、もうべきはスタミナXであろう....
本文段落
注
節
表題
精力剤の効果
本文段落
ビタミンKとスタミナXを双璧にして....
本文段落
注
続く .....

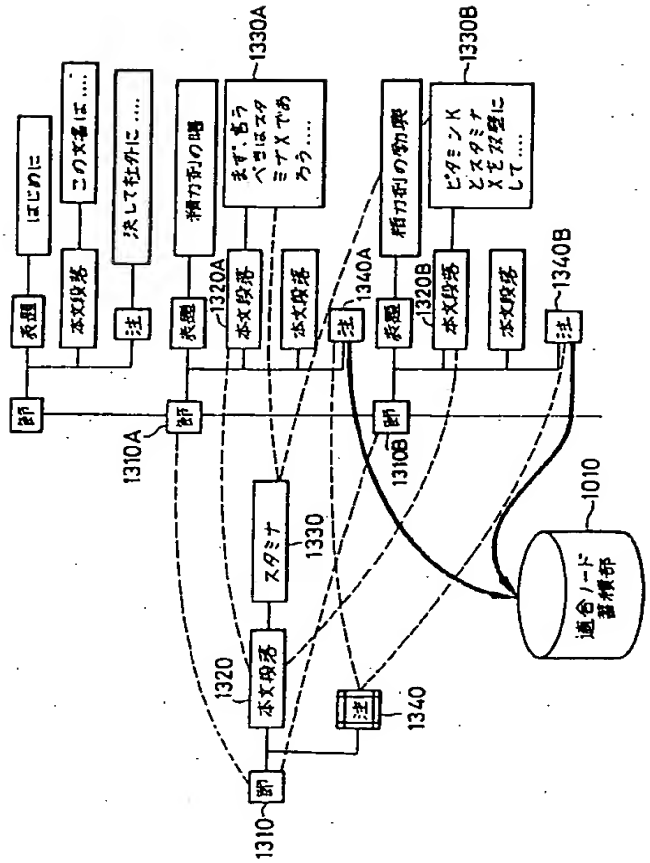
【図18】


削除情報の作成処理
親リストを空にする
1051
パターンに照合がとれた子ノードCをひとつ入力
1052
1053
親ノードPは親リストに未登録
YES
1057
NO
親リストに新規登録
1054
子ノードCは親ノードPの子リストに未登録
YES
1056
NO
子ノードCを親ノードPの子リストに新規登録
パターンに照合がとれた次の子ノードCをひとつ入力
1055
1058
NO
照合ノードは終わりか?
YES
終了

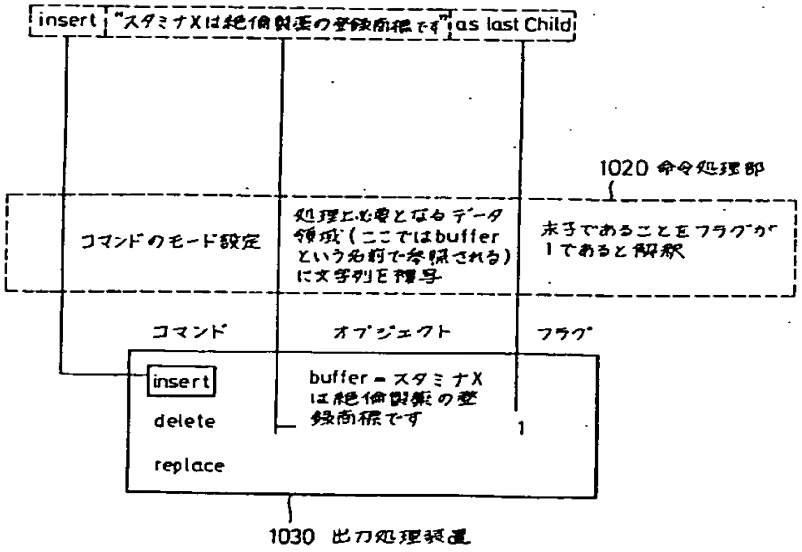
【図20】


ノード出力処理
1071
ノードNは親リストに登録
YES
NO
1074
1072
ノードNをそのまま出力
ノードNについて属性を施して出力
ノードNの全てに対してノード出力処理を実行
ノードNについての子リストに含まれるものに対して、ノード出力処理を実行
1075
1073
終了

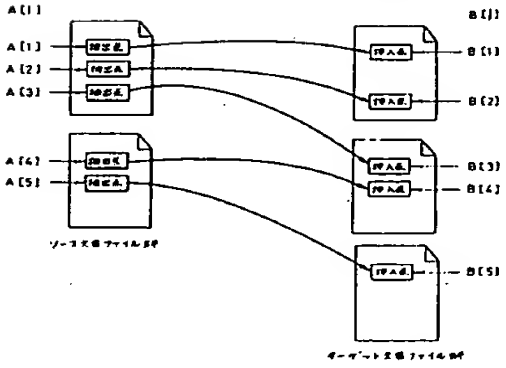
【図１３】


1330A
1330B
1320A
1320B
1340A
1340B
1310A
1310B
1320
1330
1340
1310
1010
節
表題
本文段落
注
スタミナ
通信ノード管理部

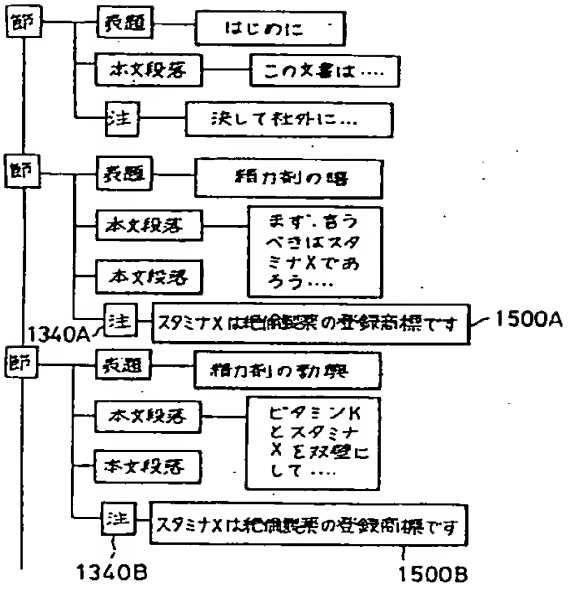
【図14】


insert
"スダミナXは絶倫関係の登録商標です"
as last Child
1020 命令処理部
コマンドのモード設定
処理上必要となるデータ領域(ここではbufferという名前で参照される)に文字列E指号
末子であることをフラグが1であると解釈
コマンド
オブジェクト
フラグ
insert
delete
replace
buffer＝スダミナXは絶倫関係の登録商標です
1
1030 出力処理装置

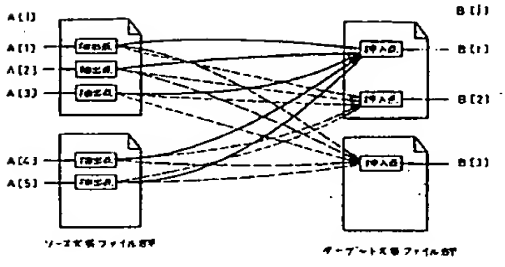
【図26】


A[1]
A[2]
A[3]
A[4]
A[5]
B[1]
B[2]
B[3]
B[4]
B[5]

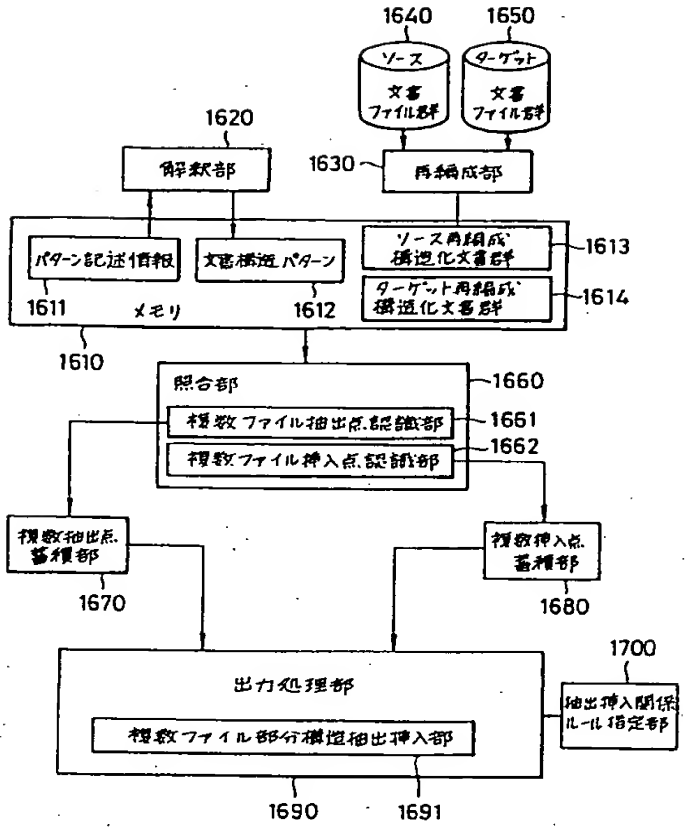
【図15】


節
表題
はじめに
本文段落
この文書は....
注
決して社外に...
節
表題
精力剤の場
本文段落
まず、吉うべきはスタミナXであろう....
本文段落
注
スタミナXは松偏製薬の登録商標です
1500A
1340A
節
表題
精力剤の効果
本文段落
ビタミンKとスタミナXを双壁にして....
本文段落
注
スタミナXは松偏製薬の登録商標です
1340B
1500B

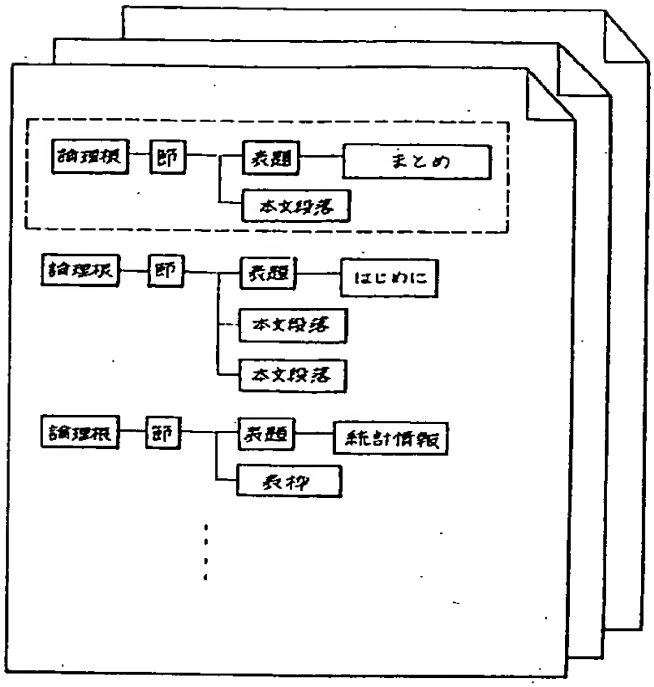
【図27】


A[1]
A[2]
A[3]
A[4]
A[5]
B[1]
B[2]
B[3]

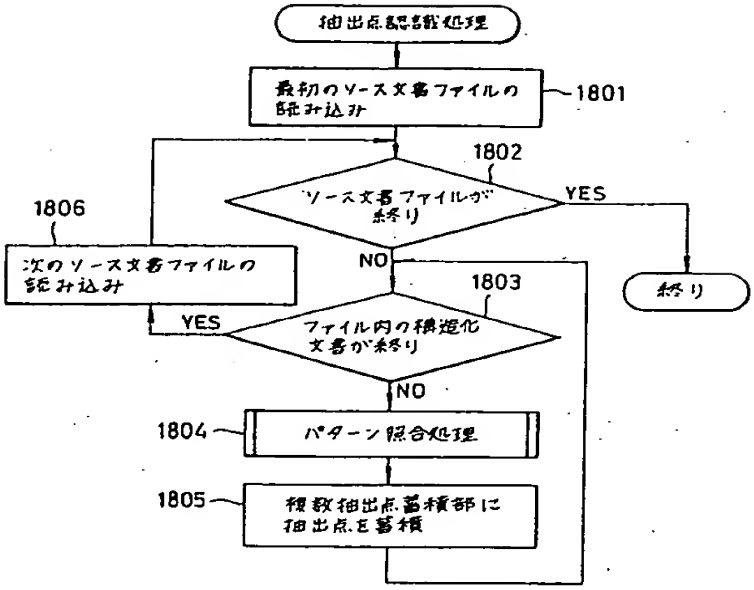
〔図21〕


1640
1650
ソース
文書ファイル群
ターゲット
文書ファイル群
1620
解釈部
1630
再編成部
パターン記述情報
文書構造パターン
ソース再編成構造化文書群
ターゲット再編成構造化文書群
1613
1614
1611
メモリ
1612
1610
照合部
1660
複数ファイル抽出点認識部
1661
複数ファイル挿入点認識部
1662
複数抽出点蓄積部
1670
複数挿入点蓄積部
1680
1700
出力処理部
抽出挿入関係ルール指定部
複数ファイル部分構造抽出挿入部
1690
1691

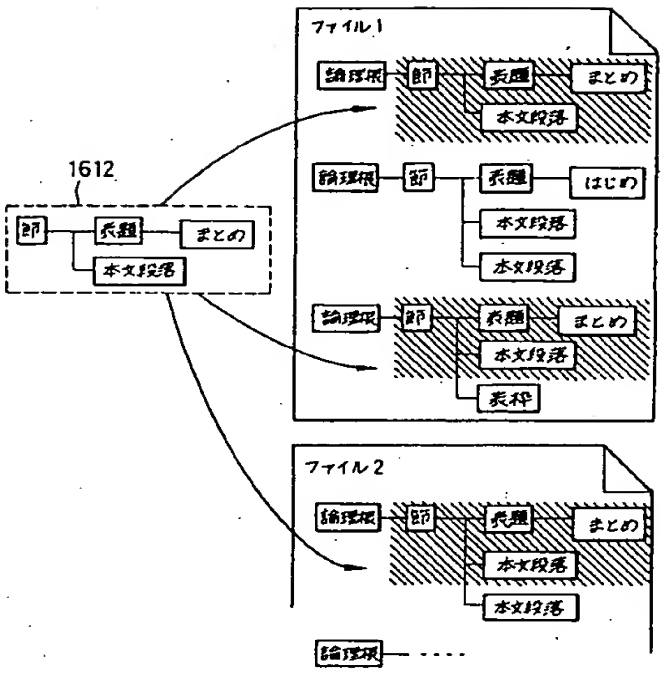
【図２２】



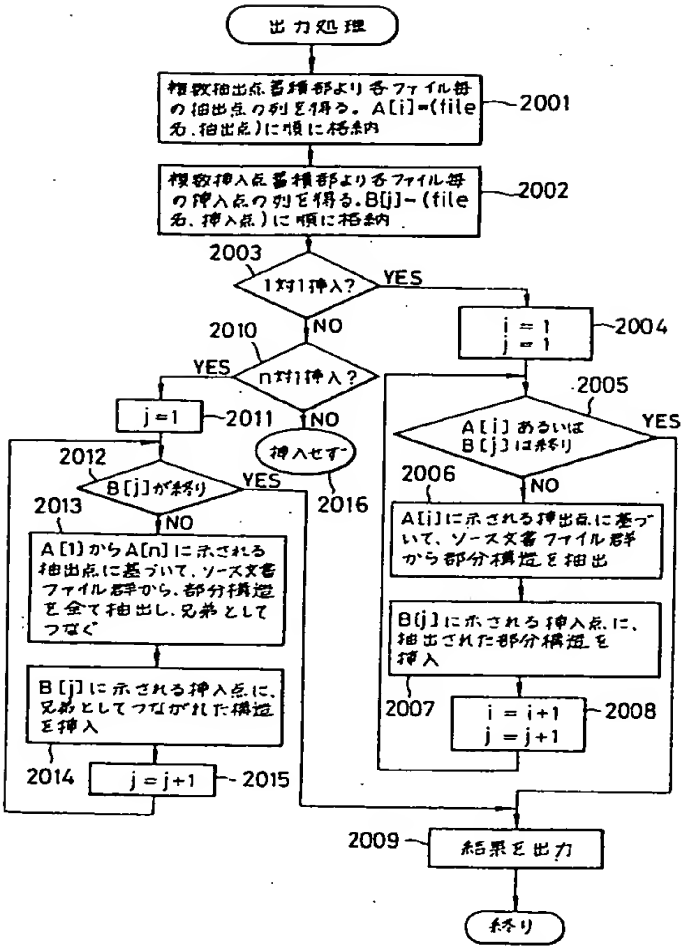
【図23】


抽出点記憶処理
最初のソース文書ファイルの読み込み
1801
1802
ソース文書ファイルが終り
YES
NO
終り
1806
次のソース文書ファイルの読み込み
YES
1803
ファイル内の構造化文書が終り
NO
1804
パターン照合処理
1805
種数抽出点蓄積部に抽出点を蓄積

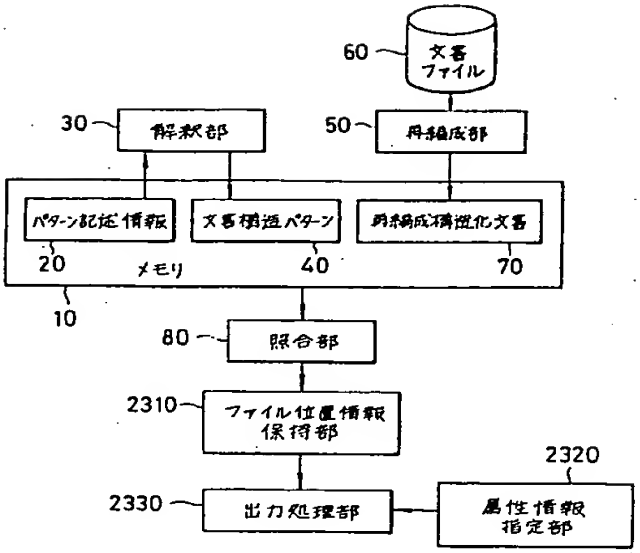
【図24】


ファイル1
論理根
節
表題
まとめ
本文段落
論理根
節
表題
はじめ
本文段落
本文段落
論理根
節
表題
まとめ
本文段落
表枠
1612
節
表題
まとめ
本文段落
ファイル2
論理根
節
表題
まとめ
本文段落
本文段落
論理根

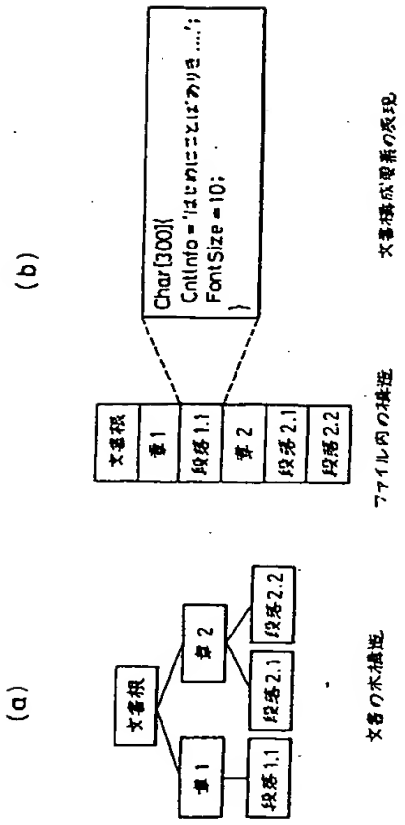
【図25】


出力処理
複数抽出点蓄積部より各ファイル毎の抽出点の列を得る。A[i]=(file名,抽出点)に順に格納
2001
複数挿入点蓄積部より各ファイル毎の挿入点の列を得る。B[j]=(file名,挿入点)に順に格納
2002
2003
1対1挿入?
YES
NO
i=1
j=1
2004
2010
n対1挿入?
YES
NO
2005
A[i]あるいはB[j]は終り
YES
NO
2006
A[i]に示される抽出点に基づいて、ソース文書ファイル群から部分構造を抽出
B[j]に示される挿入点に、抽出された部分構造を挿入
2007
i=i+1
j=j+1
2008
j=1
2011
挿入せず
2016
2012
B[j]が終り
YES
NO
2013
A[1]からA[n]に示される抽出点に基づいて、ソース文書ファイル群から、部分構造を全て抽出し、兄弟としてつなぐ
B[j]に示される挿入点に、兄弟としてつながれた構造を挿入
2014
j=j+1
2015
2009
結果を出力
終り

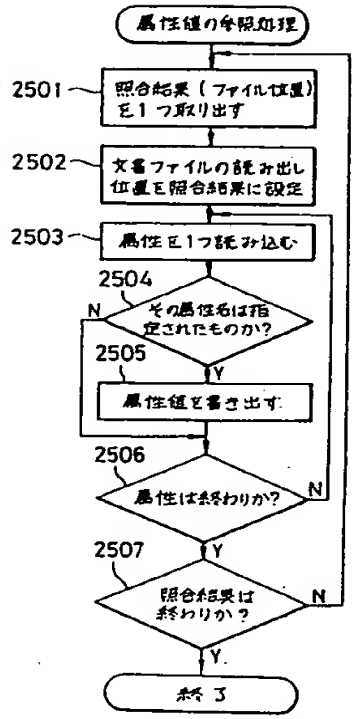
[図２８]


60
文書ファイル
30
解釈部
50
再構成部
パターン記述情報
文書構造パターン
再構成構造化文書
20
メモリ
40
70
10
80
照合部
2310
ファイル位置情報保持部
2320
2330
出力処理部
属性情報指定部

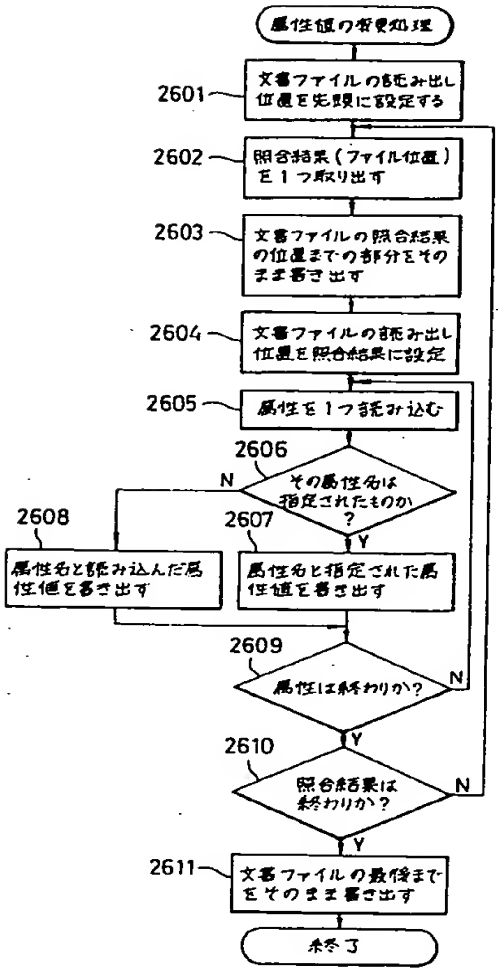
【図29】


(a)
文書の木構造
文書根
章1
章2
段落1.1
段落2.1
段落2.2
(b)
ファイルの構造
文書根
章1
段落1.1
章2
段落2.1
段落2.2
Char[300]{
CntInfo="はじめにことばありき....";
FontSize=10;
}
文書構成要素の表現

【図３０】


属性値の参照処理
2501
照合結果（ファイル位置）を1つ取り出す
2502
文書ファイルの読み出し位置を照合結果に設定
2503
属性を1つ読み込む
2504
その属性名は指定されたものか?
N
Y
2505
属性値を書き出す
2506
属性は終わりか?
N
Y
2507
照合結果は終わりか?
N
Y
終了

【図３１】


属性値の検索処理
2601
文書ファイルの読み出し位置を先頭に設定する
2602
照合結果（ファイル位置）を１つ取り出す
2603
文書ファイルの照合結果の位置までの部分をそのまま書き出す
2604
文書ファイルの読み出し位置を照合結果に設定
2605
属性を１つ読み込む
2606
その属性名は指定されたものか？
N
Y
2607
属性名と指定された属性値を書き出す
2608
属性名と読み込んだ属性値を書き出す
2609
属性は終わりか？
N
Y
2610
照合結果は終わりか？
N
Y
2611
文書ファイルの最後までをそのまま書き出す
終了

63　　64

フロントページの続き

(72)発明者　松本　天
神奈川県川崎市高津区坂戸３丁目２番１号
ＫＳＰ　Ｒ＆Ｄ　ビジネスパークビル
富士ゼロックス株式会社内